2023 IEEE Symposium Series on Computational Intelligence

December 5th – 8th 2023, Mexico City, Mexico

Proceedings of Presentation-Only Papers

SSCI 2023

IEEE

IEEE Computational Intelligence Society

Content List of 2023 IEEE Symposium Series on Computational Intelligence

Liu, Qiang	Canghai Campus，Shenzhen University, Nanshan District, She

ThC1	Imperio A
Swarm Intelligence (POP)	

10:30-10:50	ThC1.1

Spider Monkey Optimization for Optimal Operational Planning of Energy Plants, pp. 609-610.

Kobayashi, Yuto	Meiji University
Fukuyama, Yoshikazu	Meiji University
Wananabe, Takuya	Fuji Electric Co., Ltd
Iizaka, Tatsuya	Fuji Electric
Matsui, Tetsuro	Fuji Electric

10:50-11:10	ThC1.2

City Assignment by Multi-Objective Evolutionary Artificial Neural Networks for Multiple TSP, pp. 611-612.

Katada, Yoshiaki	Setsunan University
Watanabe, Shinya	Muroran Institute of Technology
Ohkura, Kazuhiro	Hiroshima University

11:10-11:30	ThC1.3

Towards Interpretable Digital Twins for Self-Aware Industrial Machines, pp. 613-614.

Santos da Silva Júnior, Adelson	University of Pernambuco
Vilar Dias, João Luiz	Universidade De Pernambuco
Buarque de Lima Neto, Fernando	University of Pernambuco

ThC2	Imperio B
Image Processing (POP)	

10:30-10:50	ThC2.1

Real Time Continuous Image Stitching Algorithm Based on SIFT, pp. 615-616.

Yang, RUIJun	Shanghai Institute of Technology
Zhang, Chu	Shanghai Institute of Technology
Cheng, Yan	East China University of Political Science and Law

10:50-11:10	ThC2.2

Synthetic Generation of Pneumonia Images Using CycleGAN Model, pp. 617-618.

Lugo Torres, Gerardo	Centro De Investigación En Computación, Instituto Politécnico Nac
Peralta, Diego Antonio	Instituto Politécnico Nacional
Valdez-Rodríguez, José E.	Centro De Investigación En Computación
Calvo, Hiram	CIC-IPN

11:10-11:30	ThC2.3

Time Series Prediction Based on Randomly Weighted Neural Networks, pp. 619-620.

Wang, Xizhao	Shenzhen University
Wang, Qin	Shenzhen University

ThC3	Imperio C
Deep Learning 1 (POP)	

10:30-10:50	ThC3.1

Seed Kernel Counting Using Domain Randomization and Object Tracking Neural Networks, pp. 621-624.

Margapuri, Venkata Siva Kumar	Villanova University
Thapaliya, Prapti	Villanova University
Neilsen, Mitchell	Kansas State University

10:50-11:10	ThC3.2

Detecting Automated Generated Text with LLMs, pp. 625-626.

Aguilar-Canto, Fernando	CIC IPN
Cardoso-Moreno, Marco A.	Cic - Ipn
Jiménez López, Diana Laura	Centro De Investigación En Computación, Instituto Politécnico Na
Calvo, Hiram	CIC-IPN

11:10-11:30	ThC3.3

Explainable Image Recognition with Graph-Based Feature Extraction and Classification, pp. 627-630.

Azam, Basim	Griffith University
Kuttichira, Deepthi	Instiute for Integrated and Intelligent Systems, Griffith Univer
Verma, Brijesh	Instiute for Integrated and Intelligent Systems, Griffith Univer

ThC4	Constitución A
Learning Algorithms (POP)	

10:30-10:50	ThC4.1

MRNA Robust Signatures for IBD Using Machine Learning, pp. 631-632.

Rojas-Velazquez, David	Utrecht University
Kidwai, Sarah	Utrecht University
de Vries, Luciënne	Division of Pharmacology, University of Utrecht,
Garssen, Johan	Division of Pharmacology, University of Utrecht
Tonda, Alberto	UMR 518 MIA-PS, INRAE, Université Paris-Saclay
Lopez-Rincon, Alejandro	Utrect University

10:50-11:10	ThC4.2

Predicting Directional Change Reversal Points with Machine Learning Regression Models, pp. 633-636.

Rayment, George	University of Essex
Kampouridis, Michael	University of Essex
Adegboye, Adesola	University of Kent

11:10-11:30	ThC4.3

Enhancing Solar Panel Efficiency through Deep Deterministic Policy Gradients (DDPG) Reinforcement Learning Control, pp.

637-638.

Ortiz-Munoz, Diana	Universidad Autonoma De Ciudad Juarez
Luviano-Cruz, David	Universidad Autonoma De Ciudad Juarez
Perez-Dominguez, Luis	Universidad Autonoma De Ciudad Juarez
Rodriguez-Ramirez, Alma	Universidad Autonoma De Ciudad Juarez

| **ThC5** | Constitución B |
| **Deep Learning 2 (POP)** | |

| 10:30-10:50 | ThC5.1 |

Simultaneous Facial Age Transformation and Reenactment, pp. 639-640.

Zhang, Jie-Ying	National Taiwan University of Science and Technology
Hsiung, Li-Syun	National Taiwan University of Science and Technology
Hsu, Gee-Sern	National Taiwan University of Science and Technology

| 10:50-11:10 | ThC5.2 |

Classification of Songs in Spanish with LLMs: An Analysis of the Construction of a Dataset, through Classification, pp. 641-642.

Alcantara, Tania	Centro De Investigación En Computación, Instituto Politécnico Na
Omar, Garcia-Vazquez	CIC-IPN
Cardoso-Moreno, Marco A.	Cic - Ipn
Calvo, Hiram	CIC-IPN

| 11:10-11:30 | ThC5.3 |

Convolutional Autoencoder-Based Multimodal One-Class Classification, pp. 643-644.

Laakom, Firas	Tampere University
Sohrab, Fahad	Tampere University
Raitoharju, Jenni Karoliina	University of Jyväskylä
Iosifidis, Alexandros	Aarhus University
Gabbouj, Moncef	Tampere University

| **ThC6** | Constitución C |
| **Automated Algorithm (POP)** | |

| 10:30-10:50 | ThC6.1 |

TransOpt: Transformer-Based Representation Learning for Optimization Problem Classification, pp. 645-646.

Cenikj, Gjorgjina	Jožef Stefan Institute
Petelin, Gašper	Jožef Stefan Institute
Eftimov, Tome	Jožef Stefan Institute

| 10:50-11:10 | ThC6.2 |

Leveraging Automation, Optimization, and Distributed Computing to Perform High-Fidelity Regional Seismic Risk and Resilience Assessment, pp. 647-648.

| Dahal, Laxman | University of California Los Angeles |
| Burton, Henry | University of California Los Angeles |

| Zhong, Kuanshi | University of Cincinnati |

| 11:10-11:30 | ThC6.3 |

Context-Based Classification of Sensitive Personal Information, pp. 649-650.

De Jesus, Sara	CIC-IPN
Aguirre Anaya, Eleazar	Instituto Politecnico Nacional
Calvo, Hiram	CIC-IPN
Coyac-Torres, Jorge E.	Centro De Investigación En Computación - IPN
Acosta Bermejo, Raúl	Instituto Politécnico Nacional

| **ThC7** | Colonia |
| **Decision Making (POP)** | |

| 10:30-10:50 | ThC7.1 |

Profit Allocation in Logistics Enterprise Coalitions Based on Fuzzy Cooperative Game Theory, pp. 651-652.

| He, Xi | Tsinghua University |
| Huang, Shuangxi | Tsinghua University |

| 10:50-11:10 | ThC7.2 |

Optimizing a Prediction-Based, Mixed-Asset Portfolio Including REITs, pp. 653-656.

| Habbab, Fatim Zahra | University of Essex |
| Kampouridis, Michael | Univ. of Essex, Essex, UK |

| 11:10-11:30 | ThC7.3 |

Computational Intelligence for Equity-Aware STEM Student Recruitment, pp. 657-658.

| Abid, Noor | University of Calgary |
| Yanushkevich, Svetlana | University of Calgary |

| **ThC8** | Conquista |
| **Data Mining (POP)** | |

| 10:30-10:50 | ThC8.1 |

Performance Comparison of Augmented Reality Frameworks, pp. 659-660.

Villagran-Vizcarra, Dafnis Cain	Universidad Autonoma De Ciudad Juarez
Luviano-Cruz, David	Universidad Autonoma De Ciudad Juarez
Perez-Dominguez, Luis	Universidad Autonoma De Ciudad Juarez

| 10:50-11:10 | ThC8.2 |

Structural Analysis of the Mexico-Toluca Interurban Train with Data Science, pp. 661-662.

Arellano, Osmar David	Universidad Autónoma Del Estado De México
Valdovinos, Rosa María	Universidad Autónoma Del Estado De México
Guzmán, Angélica	Universidad Jaume I
Delgado, David Joaquín	Universidad Autónoma Del Estado De México

| 11:10-11:30 | ThC8.3 |

Neural Network Regression for Structural Health Monitoring Using Smartphones, pp. 663-664.

| Yingqin, Zhu | CINVESTAV-IPN |

Li, Xiaoou   CINVESTAV-IPN
Ovilla-Martinez, Brisbane   CINVESTAV-IPN

15:10-15:30	FrA6.6

Hypertension and Its Relationship with Socioeconomic Factors in Mexico Using Clustering Techniques, pp. 1477-1478.

OBED SALOMON, CASILLAS   Instituto Politécnico Nacional
BALTAZAR

Pichardo-Lagunas, Obdulia   Instituto Politécnico Nacional

Martinez-Seis, Bella   IPN (UPIITA)

17:40-18:00	FrB6.6

An Ensemble Method for Applying Particle Swarm Optimization Algorithms to Systems Engineering Problems, pp. 1773-1774.

Hampshire, Ken   George Washington University

Mazzuchi, Thomas   George Washington University

Sarkani, Shahram   George Washington University

Spider Monkey Optimization
for Optimal Operational Planning of Energy Plants

Yuto Kobayashi and Yoshikazu Fukuyama
Graduate School of Advanced Mathematical Sciences
Meiji University
Tokyo, Japan
yfukuyam@meiji.ac.jp

Takuya Watanabe, Tatsuya Iizaka, and Tetsuro Matsui
Corporate R&D Headquarters
Fuji Electric Co., Ltd.
Tokyo, Japan

Abstract—This paper proposes a spider monkey optimization (SMO) based method for optimal operational planning of energy plants (OOPEPs). The effectiveness of the proposed spider monkey optimization (SMO) based method is verified by comparing with the differential evolutionary particle swarm optimization (DEEPSO), brain storm optimization (BSO), modified BSO (MBSO), and multi-population MBSO (MP-MBSO) based methods. The results are verified by the statistical tests.

Keywords—Optimal operational planning of energy plants, mixed integer non-linear programming problem, spider monkey optimization, cooperative metaheuristics

I. Introduction

In recent years, Building/Factory Energy Management System (BEMS/FEMS) have drawn attention. BEMS and FEMS are utilized in large-scale office buildings, commercial facilities, and factories. One of the purposes of BEMS and FEMS is to optimize the operation of energy plants in a facility based on the energy consumption of the target facility. The realization of optimal operation of the energy plant in a facility contributes to a reduction of purchased energy costs and has a significant management impact. Therefore, research has been conducted on optimal energy plant operational planning for BEMS and FEMS [1]-[4].

In the OOPEP, it is required to simultaneously consider both linear and nonlinear equipment characteristics, binary variables for start-stop status, and continuous variables for input/output quantities for each piece of equipment in the energy plant. Thus, the problem is formulated as a mixed-integer nonlinear programming problem.

This paper proposes a SMO based method for the OOPEPs. SMO is one of the methods that dynamically changes the number of groups according to the search situation. The effectiveness of the proposed method was confirmed by comparing it with DEEPSO, BSO, MBSO, and MP-MBSO based methods. The results are verified using the Kruskal-Wallis test and the Mann-Whitney U tests with holm correction.

II. A Problem Formulation of The OOPEPs

A. A Target Energy Plant Model

The target energy plant model in this paper which is a benchmark proposed by the Institute of Electrical Engineers of Japan (IEEJ) as a model of energy plants installed in large commercial facilities and factories is shown in Fig.1 [5].

B. Decision Variables

(a) Gas turbine generators (GTGs): start-stop status($y_{gk}^i \in \{0,1\}: i = 1, \dots, T, k = 1, \dots, N_g$), natural gas input quantities ($x_{gk}^i: i = 1, \dots, T, k = 1, \dots, N_g$).

(b) Turbo Refrigerators (TRs): start-stop status ($y_{tk}^i \in \{0,1\}: i = 1, \dots, T, k = 1, \dots, N_t$), heat output quantities ($x_{tk}^i: i = 1, \dots, T, k = 1, \dots, N_t$).

(c) Steam Refrigerators (SRs): start-stop status ($y_{sk}^i \in \{0,1\}: i = 1, \dots, T, k = 1, \dots, N_s$), heat output quantities ($x_{sk}^i: i = 1, \dots, T, k = 1, \dots, N_s$).d

T is the 24 hours of the day ($T = 24$), N_g is the number of GTGs, N_t is the number of TRs, N_s is the number of SRs. When the binary variable y^i is 0, the equipment is stopped, and when y^i is 1, the equipment is in operation.

C. Objective Function

The objective function is to minimize sum of the costs (purchased electric power and purchased natural gas) required to operate the energy plant 24 hours a day.

D. Constraints

1) Supply and demand constraints: The quantity of each energy supply, such as electricity, steam energy, and thermal energy, must be equal to the quantity of each energy consumed at each time of day.

2) Facility constraints: The mechanical characteristics and operational limits of each piece of equipment that composes the energy plant are presented by benchmark problem.

3) Operational constraints: Each equipment has a minimum continuous operation time and continuous shutdown time.

III. OOPEPs Using Spider Monkey Optimization

A. A Round-up Function [6]

OOPEPs require simultaneous consideration of both binary and continuous variables. In this paper, by applying a round-up function, the binary variables can be treated as
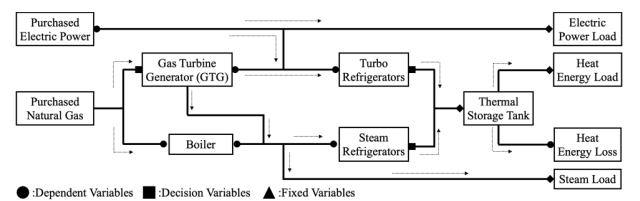

Fig.1. A target energy plant model.

continuous variables. Using the function, it is expected that a search for the shutdown status of each piece of equipment will be sufficiently performed.

B. Spider Moneky Optimization

SMO is a dynamic multi-population based multi-point search evolutionary computation method proposed by J. C. Bansal et al. in 2014 [7]. The method was proposed to simulate the fission-fusion social structure of spider monkeys. They usually form large groups, and depending on the situation, such as foraging, large one group is divided into several smaller groups led by a particular spider monkey to feed.

C. OOPEPs using Spider Monkey Optimization

The overview of algorithm of the proposed SMO based method applied to OOPEPs is following.

Step.1 Initialize each individual (SMs) using the round-up function. $iter = 1$.

Step.2 Update each individual using Local Leaders (LLs) and Global Leader (GL). After converting the decision variable values into the input/output values and operating status of each equipment based on the round-up function, the objective function values of each individual are calculated.

Step.3 Update LLs and GL based on the updated population.

Step.4 Random search and reconstruct the population based on the update cycle of LLs and GL.

Step.5 Update LLs and GL. If $iter$ reaches its maximum, output the best solution so far and exit. If $iter$ is not reached, $iter = iter + 1$, go to Step 2.

IV. SIMULATIONS

Effectiveness of the proposed method is verified by applying the DEEPSO based method [2], the BSO based method [3], the MBSO based method [4], the MP-MBSO based method [4] and the proposed SMO based method to the energy plant model shown in Figure 1.

A. Simulation Conditions

The energy plant of the model consists of one GTG, one TR, two SRs, and one boiler. The parameters for the methods are shown below.

1) Common parameters:the maximum number of evalutions: 320,000 (The value is determined so that the evaluation number of all methods are the same.), *the number of trials:* 100.

2) The round-up function's parameters:
$\alpha: -0.5, \beta: -0.05, \gamma: 1.05$

3) The proposed SMO based method's parameters: The maximum number of N_g: 5, the number of individuals SMs: 40, *probability threshold pr: 0.8, the maximum number of LLC*: 1, *the maximum number of GLC*: 5

B. Simulation Results

Table I shows the minimum, the maximum, average, and standard deviation values of the objective function values by the proposed SMO based method, and the conventional methods and the p-value of the Kruskal-Wallis test. Each value in Table I represents the value when the average value by DEEPSO based method is set to 100 as the benchmark. These simulation results indicate that the proposed SMO based method is the best in all values. The results of the

Kruskal-Wallis test verified to be a significant difference at 5% significance level.

Moreover, to reveal significant differences among the methods, the Mann-Whitney U tests with holm correction are conducted as a post hoc test, and the results are shown in Table II. In Table II, the p-values between the proposed method and other conventional methods are not exceeded 0.05 significant level. Therefore, it is verified that the proposed SMO based method is superior to the conventional DEEPSO, BSO, MBSO, MP-MBSO based methods.

V. CONCLUSIONS

This paper proposes the spider monkey optimization based method for the OOPEPs. It was confirmed by simulation using a plant model that the proposed method can stably obtain better quality solutions compared to the conventional methods. Moreover, effectiveness of the proposed method is verified by the statistical tests.

As future works, applications of the improved spider monkey optimization methods and other advanced evolutionary computation methods to the OOPEPs will be investigated.

REFERENCES

[1] T. Tsukada, et al., "Optimal Operational Planning for Cogeneration System Using Particle Swarm Optimization," Proceedings of IEEE SIS, pp.138-143, April 2003.

[2] N. Nishimura, et al., "Optimal Operational Planning of Energy Plants by Differential Evolutionary Particle Swarm Optimization," Proceedings of IEEE POWERCON, October 2016.

[3] K. Arai, et al., "Brain Strom Optimization for Optimal Operational Planning of Energy Plant," Proceedings of Annual Conference of IEEJ, No.3-025, March 2018 (in Japanese).

[4] K. Arai, et al., "Multi-population Modified Brain Storm Optimization for Optimal Operational Planning of Energy Plants," Proceedings of IEEE International Conference on Systems, Man, and Cybernetics (SMC), pp. 935-940, October 2018.

[5] Investigating R&D Committee on New Development of Computational Intelligence Techniques and Their Applications to Industrial Systems, "Optimization benchmark problems for industrial applications," IEEJ Tech. Report, No. 1287, July 2013 (in Japanese).

[6] T. Okamoto, et al., "Application examples of energy plant operation plan problems and optimization methods," Proceedings of Annual Conference of IEEJ, 4-S21-6, March 2014 (in Japanese).

[7] J. C. Bansal, et al., "Spider Monkey Optimization algorithm for numerical optimization," Memetic Computing, Vol.6, pp.31-47, January 2014

TABLE I. THE COMPARITON OF THE MINIMUM, THE MAXIMUM, AVERAGE, AND STANDARD DEVIATION VALUES OF THE OBJECTIVE FUNCTION VALUES BY THE PROPOSED SMO BASED METHOD AND THE CONVENTIONAL DEEPSO, BSO, MBSO, MP-MBSO BASED METHODS.

	MIN.	MAX.	AVE.	STD.	p-value
DEEPSO[2]	98.42	102.63	100.00	100.00	
BSO[3]	98.20	99.71	98.74	23.93	
MBSO[4]	98.12	101.95	99.93	85.17	**4.67E-111**
MP-MBSO[4]	97.36	97.75	97.47	6.31	
SMO	**97.23**	**97.41**	**97.29**	**3.04**	

TABLE II. P-VALUES OF THE MANN-WHITENEY U TEST WITH HOLM CORRECTION BETWEEN EACH TWO METHODS..

	DEEPSO	BSO	MBSO	MP-MBSO	SMO
DEEPSO[2]	-	1.14E-20	9.85E-01	2.56E-33	2.56E-33
BSO[3]	S	-	4.72E-20	2.56E-33	2.56E-33
MBSO[4]	NS	S	-	2.56E-33	2.56E-33
MP-MBSO[4]	S	S	S	-	2.56E-33
SMO	S	S	S	S	-

*S: Significant (the p-value is not exceeded 0.05 significant level.)
NS: Non-Significant (the p-value is exceeded 0.05 significant level.)

mRNA Robust Signature for IBD Using Machine Learning

1ˢᵗ David Rojas-Velazquez
Division of Pharmacology, University of Utrecht,
Department of Data Science,
Julius Center for Health Sciences
and Primary Care, University Medical
Center Utrech The Netherlands
e.d.rojasvelazquez@uu.nl

2ⁿᵈ Sarah Kidwai
Division of Pharmacology,
University of Utrecht
Utrecht, The Netherlands
s.kidwai@uu.nl

3ʳᵈ Luciënne de Vries
Division of Pharmacology,
University of Utrecht
Utrecht, The Netherlands
l.devries14@students.uu.nl

4ᵗʰ Johan Garssen
Division of Pharmacology,
University of Utrecht,
Global Centre of Excellence
Immunology Danone Nutricia Research
Utrecht, The Netherlands
J.Garssen@uu.nl

5ᵗʰ Alberto Tonda
UMR 518 MIAPS, INRAE,
Université Paris-Saclay, Institut
des Systèmes Complexes de Paris
ÎledeFrance (ISC-PIF)
- UAR 3611 CNRS
Paris, France
alberto.tonda@inrae.fr

6ᵗʰ Alejandro Lopez-Rincon
Division of Pharmacology,
University of Utrecht,
Department of Data Science,
Julius Center for Health Sciences
and Primary Care, University Medical
Center Utrecht The Netherlands
a.lopezrincon@uu.nl

Abstract—**Inflammatory bowel disease, including Crohn's disease and ulcerative colitis, is a rising global issue. Accurate diagnosis is vital but challenging. This study used the REFS algorithm to identify IBD biomarkers using three mRNA datasets from the GEO repository. The selected genes demonstrated excellent diagnostic accuracy, highlighting the potential of machine learning in advancing IBD research.**

Index Terms—**REFS, biomarkers, IBD, mRNA processing, bioinformatics**

I. INTRODUCTION

Inflammatory bowel disease (IBD) is a rising global health issue. It includes Crohn's disease, which can cause inflammation anywhere in the gastrointestinal tract, and ulcerative colitis, which only affects the colon's mucosal layer [1]. Accurate diagnosis is difficult due to various clinical factors. Delayed or incorrect diagnosis can worsen the disease and complicate remission [2]. New high-throughput technologies allow omics data analysis, using bioinformatics tools and machine learning methods, to achieve precision medicine goals. These tools are promising for identifying clinically relevant patterns and predictive markers for complex diseases like IBD [2]. In this study, we used the REFS algorithm [4], [5] on three mRNA datasets from the GEO repository to identify potential IBD biomarkers.

II. MATHERIALS AND METHODS

A. mRNA datasets

The three datasets used in this study were downloaded from the Gene Expression Omnibus (GEO) repository: 1) accession number GSE3365 [3] (127 samples: 85 IBD / 42 controls), 2) accession number GSE71730 [6] (47 samples: 37 IBD / 10 controls), and 3) accession number GSE33943 [7] (58 samples: 45 IBD / 13 controls). Samples were categorized into two groups: control (0) and IBD (1), which includes CD and UC patients. The GSE3365 dataset, having the most samples, was chosen for discovery.

B. REFS

The Recursive Ensemble Feature Selection (REFS) algorithm is used to identify biomarkers by determining which features are most effective in distinguishing between case and control groups [4], [5]. The ensemble is composed by 8 classifiers from the scikit-learn toolbox [8]: Stochastic Gradient Descent (SGD) on linear models, Support Vector Machine classifier (SVC), Gradient Boosting, Random Forest, Logistic Regression, Passive Aggressive classifier, Ridge Classifier and Bagging. REFS uses a nested-cross validation within a 10-fold scheme, ensuring accurate and unbiased results [9], and a validation module to minimize gene selection bias with five additional classifiers from the scikit-learn toolkit [8]: AdaBoost, Extra Trees, KNeighbors, MLP, and LassoCV. The AUC metric, derived from the average accuracy of five classifiers in a nested 10-fold cross-validation, measures the effectiveness of a discriminant test. Near 1.0 values signify high performance [10].

III. RESULTS

Using the GSE3365 dataset, REFS identified 16 out of 22,283 genes as the most effective for distinguishing IBD patients. With these genes, REFS achieved its highest accuracy
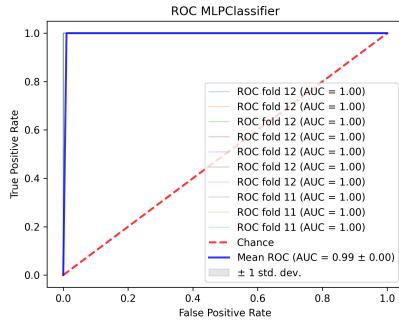
Fig. 1. Individual AUC for the classifier with the best performance in the validation process (MLP) was calculated using the 16 genes selected by REFS from GSE3365.
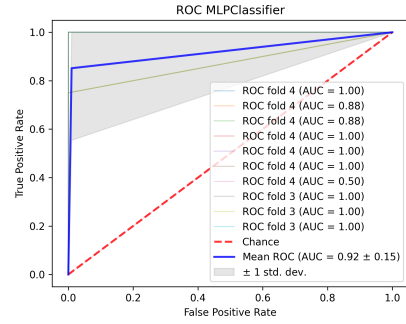

Fig. 2. Individual AUC for the classifier with the best performance in the validation process (MLP) was calculated using the 16 genes selected by REFS in the GSE71730 test database.
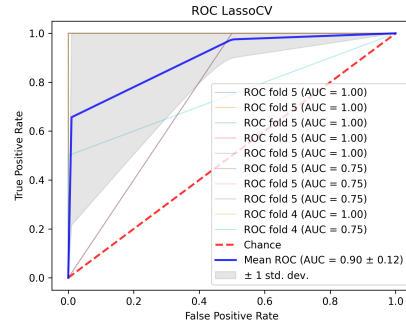

Fig. 3. Individual AUC for the classifier with the best performance in the validation process LassoCV was calculated using the 16 genes selected by REFS in the GSE33943 test database.

(over 0.97) in its feature selection module. The validation module in REFS applied to the 16 selected genes yielded an average AUC of 0.99, the classifier with the best performance was the Multi-Layer Perceptron (MLP) with an individual AUC of 0.99, see Fig. 1. The resulting average AUC corresponds to a "*excellent*" diagnostic accuracy [10]. The efficiency of the 16 genes selected by REFS was compared against SelectKBest algorithm ($k = 16$) from the scikit-learn toolbox [8], and a 10-times repeated random selection. The average AUCs were 0.946 for SelectKBest and 0.7194 for random selection. The genes with higher AUCs chosen by REFS were then tested on two datasets: GSE71730 and GSE33943.

After identifying the 16 genes selected by using REFS in GSE71730 and GSE33943 testing datasets, REFS validation yielded an average AUC of 0.874 and 0.894 respectively. Both AUC correspond to "*very good*" diagnostic accuracy [10]. The classifiers with the best performance were MLP with an individual AUC of 0.92 for GSE71730, see Fig. 2, and LassoCV with an individual AUC of 0.90 for GSE33943, see Fig. 3. A 10-time random validation was done on both testing datasets, randomly selecting 16 genes each time. REFS validation was applied, the resulting average AUCs for GSE71730 was 0.66 and 0.7472 for GSE33943. Finally, after identifying the 16 genes selected by using SelectKBest in GSE71730 and GSE33943 testing datasets, REFS validation yielded an average AUC of 0.784 and 0.862 respectively.

IV. CONCLUSION

This study used the REFS algorithm to identify potential IBD biomarkers from three mRNA databases. One database was used for discovery and the others for testing. The selected genes by using REFS showed better diagnostic accuracy compared by those selected by using SelectKBest and 10-time random validation. Despite limited information on these genes' impact on IBD, the approach can yield promising results for IBD diagnosis and treatment development.

REFERENCES

[1] Y. Z. Zhang, and Y. Y. Li, "Inflammatory bowel disease: pathogenesis," World journal of gastroenterology: WJG, 20(1), 2014, pp. 91–99. doi.org/10.3748/wjg.v20.i1.91.

[2] B. Stankovic, N. Kotur, G. Nikcevic, V. Gasic, B. Zukic, and S. Pavlovic, "Machine learning modeling from omics data as prospective tool for improvement of inflammatory bowel disease diagnosis and clinical classification," Genes, 12(9), 2021, p. 1438. doi: 10.3390/genes12091438.

[3] M.E. Burczynski, et al., "Molecular classification of Crohn's disease and ulcerative colitis patients using transcriptional profiles in peripheral blood mononuclear cells," The journal of molecular diagnostics, 8(1), 2006, pp. 51–61. doi.org/10.2353/jmoldx.2006.050079.

[4] A. Lopez-Rincon, M. Martinez-Archundia, G. U. Martinez-Ruiz, A. Schoenhuth, and A. Tonda, "Automatic discovery of 100-miRNA signature for cancer classification using ensemble feature selection," BMC bioinformatics, 20, 2019, pp. 1–17. doi.org/10.1186/s12859-019-3050-8.

[5] K. Kamphorst, A. Lopez-Rincon, A. M. Vlieger, J. Garssen, E. van't Riet, and R. M. van Elburg, "Predictive factors for allergy at 4–6 years of age based on machine learning: A pilot study," PharmaNutrition, 23, 2023, p. 100326. doi.org/10.1016/j.phanu.2022.100326.

[6] B. Gurram, et al., "Plasma-induced signatures reveal an extracellular milieu possessing an immunoregulatory bias in treatment-naive paediatric inflammatory bowel disease," Clinical and Experimental Immunology 184(1), 2016, pp. 36–49. doi.org/10.1111/cei.12753.

[7] P. P. E. van Lierop, et al., "Gene expression analysis of peripheral cells for subclassification of pediatric inflammatory bowel disease in remission," PLoS One 8(11), 2013, p. e79549. doi.org/10.1371/journal.pone.0079549.

[8] F.Pedregosa, et al., "Scikit-learn: Machine learning in Python," the Journal of machine Learning research, 12, 2011, pp. 2825-2830.

[9] A. Vabalas, et al, "Machine learning algorithm validation with a limited sample size," PloS one 14(11), 2019, p. e0224365. doi.org/10.1371/journal.pone.0224365.

[10] A. M. Šimundić, "Measures of diagnostic accuracy: basic definitions," ejifcc, 19(4), 2009, p. 203.

Predicting Directional Change Reversal Points with Machine Learning Regression Models

George Rayment, Michael Kampouridis
School of Computer Science and Electronic Engineering
University of Essex
Wivenhoe Park, United Kingdom
{gr17754, mkampo}@essex.ac.uk

Adesola Adegboye
School of Computing
University of Kent
Medway, United Kingdom
adesolaadegboye@gmail.com

Abstract—Traditional trading methods often use fixed-interval sampling to capture price changes. In this work, we use an intrinsic time sampling method referred to as directional changes (DC), which reports information whenever there is a significant price change. Tick data from an array of seven FX currency pairs is sampled using the DC framework. We then compare eleven different machine learning (ML) algorithms in a regression task of predicting when the current trend in the market will reverse. These algorithms are: decision tree, random forest, support vector regression, linear regression, stochastic gradient descent regression, kernel ridge regression, elastic net regression, bayesian ridge regression, gradient boosting regression, multilayer perceptron, and long short-term memory neural network. Predicting trend reversal is crucial in trading, as it allows us to anticipate changes in the market and take the relevant actions that are necessary to maximise our returns. After identifying the best ML algorithm for a dataset, we use this prediction as an input of a DC-based trading strategy, and report its performance in terms of return and risk (maximum drawdown). We also benchmark this strategy against four other trading strategies, which include technical analysis and buy and hold. Results over 349 datasets show that the proposed DC-based trading strategy is able to consistently offer high returns at low risk, statistically and significantly outperforming all other benchmarks.

Index Terms—machine learning, regression, directional changes, high-frequency trading

I. INTRODUCTION

The foreign exchange (FX) market is an electronic platform upon which one currency can be exchanged for another. Predicting FX price movement is an important problem in finance that has attracted many researchers due to it's complex nature and opportunity for participants to increase profits at reduced risk. Modern approaches to predicting price movement often involve generating financial indicators from historical data and training a machine learning (ML) model to predict the future price movement. Traditionally, the indicators were derived from tick prices sampled at fixed intervals. Different techniques have been proposed in the literature for creating intrinsic time series such as directional changes (DC) [1], which is the chosen technique in this work. The motivation for using this technique stems from the basis that it has the concept of confirming the existence of a trend whilst the trend is ongoing.

In the directional changes approach, an event summary is generated by recording alternating upward and downward directional changes trends according to a threshold θ. Each trend is subdivided into a directional change (DC) event and an overshoot (OS) event. The DC event is the event that caused a significant price shift either upward or downward.

The OS event is an event between two adjacent DC events representing the time between when a DC trend is first confirmed and when a new trend, in the opposite direction begins. The occurrence of a DC trend is confirmed whilst in the trend howbeit in hindsight because its DC event's length is deduced easily whilst the OS event's length is determined after the next DC trend in the opposite direction is confirmed. One of the challenges in the DC approach is the accurate estimation of the OS event length. In this work, we leverage the inherent strength of eleven well-known ML algorithms, and design a multi-ML algorithm trading framework for predicting DC trend reversal.

II. DIRECTIONAL CHANGES - BACKGROUND

Directional changes is a data sampling technique used in creating intrinsic time-series from a physical time-series. First, a threshold value θ that expresses a significant change in price is predetermined by a trader. Successive alternating snapshots of the market are then recorded when a change in price is equal to or greater than the threshold, creating a time-series that obfuscates noise between adjacent snapshots.

Figure 1 presents a sample physical time-series converted into DC event series. From the figure, each snapshot, a combination of solid and broken lines is a trend. A DC trend can either be upward or downward. Adjacent blue lines represents downward trends and adjacent red lines represent up trends. The DC trend is composed of a directional change (DC) event (i.e. a solid line) and an overshoot (OS) event (i.e. a broken line). A directional changes confirmation (DCC) point is the moment in time when price is observed to be greater than a given threshold and demarcates the DC event from the OS event. The end of an OS event is known as a directional change extreme (DCE) point. It is determined in hindsight, after the next DC event in the opposite direction is confirmed.

III. LITERATURE REVIEW

The concept of transforming physical time series into DC event series was first introduced in 1997 by [1], which empirically formalised 12 DC-based scaling laws using high-frequency data from 13 major FX markets. Subsequent works, such as [2] and [3] formalised additional scaling laws. Other works, such as [4], [5] proposed new DC-based indicators, which were used for various tasks, e.g. profiling data, and identifying regime changes.

ML is a popular technical approach for building DC based trading systems, with evolutionary techniques offering the
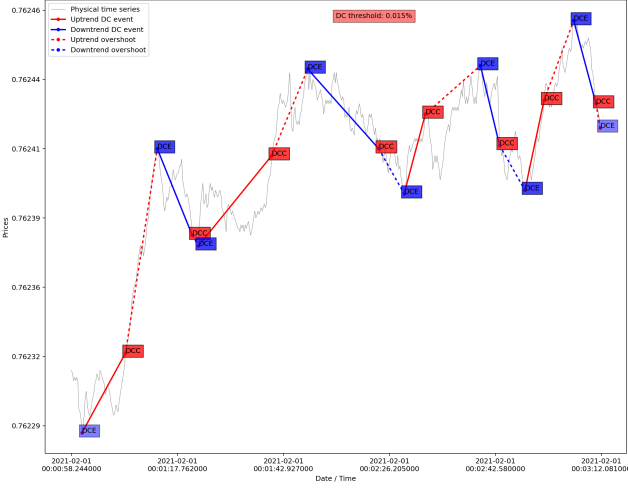
Fig. 1: Directional Change Sampling Diagram of AUDUSD

deepest body of work in conjunction with DC sampling. Works such as [6] and [7], use genetic programming (GP) to build trading strategies. In [6] two GP algorithms were built, one to trade using the DC framework and the other using physical time. The DC framework outperformed the physical time-based algorithm. GP can also be used to solve regression problems, as done in [7] where a genetic programming model is used to conduct a regression task to predict trend reversal points in DC event series.

Genetic algorithms (GAs) have also been used to develop trading strategies in DC event-series [8] and [9]. [8] used GAs to generate trading strategies that outperform technical analysis and Buy and Hold strategies in the FX market. The strategies developed using the DC framework and GAs were shown to outperform the benchmark Buy and Hold and physical time-based technical analysis strategies. [9] used the DC framework to create strategies which were then optimised using GAs. The four DC-based trading strategies introduced in this paper significantly outperformed the benchmark strategies.

The combination of DC event series created from high-frequency FX data and ML techniques, in general, has offered a successful approach to creating trading strategies. However, after conducting the above literature review, it can be deduced that the intersection of the DC event series and evolutionary techniques has been explored to a level of greater depth in comparison to other ML techniques.

IV. METHODOLOGY

Our methodology consists of a regression task and a trading task. In the regression task, for a given dataset, we first transform the physical time series (tick data) to DC events using a DC threshold (θ) of 0.015%. We then construct the features and the target variable using the start and end price of the DC move and DC indicators (N_{DC}, C_{DC}, A_T, OSV, T_{DC}, and R_{DC} (see Table I) over various periods as shown in Table II [6]). The end of the OS move is then used as the target variable in the regression task. Once the features and target table have been created, we split the data into training, validation and test sets. We then apply the following eleven ML algorithms: Decision Tree (DT), Random Forest (RF),

Support Vector Machine (SVM), Linear Regression (LR), Stochastic Gradient Descent Regression (SGD), Kernel Ridge Regression (KR), Elastic Net Regression (EN), Bayesian Ridge Regression (BR), Gradient Boosting Regression (GB), Multilayer Perceptron neural network (MLP) and a Long Short-Term Memory neural network (LSTM). The algorithm that returns the model with the lowest mean absolute percentage error (MAPE) (see Equation 1) in the validation set is then embedded into a DC-based trading strategy.

$$MAPE = \frac{1}{n} \sum_{i=1}^{n} \frac{|y_i - \hat{y}_i|}{y_i} \qquad (1)$$

TABLE I: DC Indicators
Where: θ is DC threshold and DCC is DC confirmation point

Indicator	Explanation	Equation
TMV	Ratio of whole price move to threshold	$\frac{|\Delta price|}{\theta}$
OSV	Percentage change between current DCC and previous DCC normalised by threshold	$\frac{\left(\frac{DCC_t - DCC_{t-1}}{DCC_{t-1}}\right)}{\theta}$
R_{DC}	Number of ticks adjusted by the return of the event	$\frac{(TMV*\theta)}{\Delta Event_t}$
T_{DC}	Number of ticks over the course of the event	$\Delta Event_{no.ticks}$
N_{DC}	Number of ticks over a certain number of events	$\sum_{i=0}^{n} Event_{no.ticks_i}$
C_{DC}	Sum of $|TMV|$ over a certain number of events	$\sum_{i=0}^{n} |TMV|_i$
A_T	Difference between the number of ticks spent on an up and down trend over a certain number of events	$\sum_{i=0}^{n} UpEvent_{no.ticks_i} - \sum_{i=0}^{n} DownEvent_{no.ticks_i}$

TABLE II: Feature Set

Indicator	Period
TMV	-
OSV	-
$AverageOSV$	(3, 5, 10)
R_{DC}	-
$AverageR_{DC}$	(3, 5, 10)
T_{DC}	-
$AverageT_{DC}$	(3, 5, 10)
N_{DC}	(10, 20, 30, 40, 50)
C_{DC}	(10, 20, 30, 40, 50)
A_T	(10, 20, 30, 40, 50)

In the trading task we define a trading algorithm which we call MLDC (see Algorithm 1). In this strategy we open a position (buy or sell) when the predicted OS end price is beyond the DCC by the threshold θ. We close the position when either the predicted price or the next DCC is reached, whichever occurs first. We also apply a drawdown threshold from the initial balance which will end trading if it is reached to minimise losses. Trading performance is measured per pair for a whole year using total return, maximum drawdown and Calmar ratio. Total return and maximum drawdown give a breakdown of how well the model performs and the risk associated with this performance, the Calmar ratio is then used to represent both these performance metrics in a single

value and therefore provide a measure of risk adjusted return. These results are then compared to the trading benchmarks using the same set of performance metrics.

Algorithm 1 MLDC Trading Algorithm

Initialisation: $B, B_{initial} \leftarrow 100$ // Initialise Balance
1: **for** DCC **do**
2:     Obtain OS_{pred} // Predicted price where OS event ends
3:     **if** $event$ is $Upturn$ **then**
4:         **if** $OS_{pred} \geq DCC + \theta$ **then**
5:             Enter buy position with whole balance
6:         **end if**
7:     **else if** $event$ is $Downturn$ **then**
8:         **if** $OS_{pred} \leq DCC - \theta$ **then**
9:             Enter sell position with whole balance
10:         **end if**
11:     **end if**
12:     **if** Next $event$ occurs before OS_{pred} **then**
13:         Close Position at Next DCC
14:     **else**
15:         Close Position at OS_{pred}
16:     **end if**
17:     **if** $B \leq 0.001 \times B_{initial}$ **then**
18:         Stop Trading
19:     **end if**
20: **end for**

For the regression task the results were compared to a benchmark using the $OS \approx 2DC$ scaling law [1]. For the trading task we compared the MLDC strategy to four different benchmarks AvgOS, B%H, 2DC and TA which include technical analysis, buy and hold and scaling law based trading algorithms (see Section V-B for details).

V. EXPERIMENTAL SETUP

A. Data

The tick data for the seven currency pairs is downloaded from TrueFX.com[1] and are taken from the period of 01/02/2021 to 31/01/2022. The seven currency pairs all include USD (AUD/USD, EUR/USD, GBP/USD, NZD/USD, USD/CAD, USD/CHF and USD/JPY). The DC sampling method is applied to the midpoint of the bid and ask prices obtained in the raw data. This sampling method produces a new dataset of contiguous events under the DC framework using a DC threshold (θ) of 0.015% in order to provide enough events to generate reliable results over a large amount data. This fixed threshold is selected after an empirical experimentation process shows this threshold value provides the most information rich dataset.[2]

B. Benchmarks

a) Scaling Law Regression: This regression benchmark uses the ($OS \approx 2DC$) scaling law to develop a regression benchmark that allows us to observe if the ML algorithms

---

[1] https://www.truefx.com/truefx-historical-downloads/
[2] Different techniques have been used in the literature to determine an appropriate threshold but in order to test the ability of the novel trading system we have developed, we limited the scope of this paper to a fixed threshold. Future work can be undertaken to develop on this approach with other threshold values.

---

are able to produce a more effective regression model than the empirical evidence of such scaling laws.

b) AvgOS: The AvgOS strategy calculates the average number of ticks in the OS events in the training set and then uses this value to predict the end of the OS (and thus trend reversal) on each event in the test set.

c) Buy and Hold (B&H): The B&H strategy enters a long position on the first event and then exits that position on the final event, making a single trade over the duration of the data.

d) 2DC: The 2DC strategy is the trading implementation of the Scaling Law regression method mentioned in Section V-B0a. A position is entered in the direction of a trend at the DCC point. This trade is then exited when the number of ticks since the DCC point is twice that of those during the DC move of the trend.

e) TA: This strategy calculates two MAs, namely a fast and a slow MA, derived from the tick prices for two separate periods. When the fast MA crosses above the slow MA, a buy signal is given and conversely, when it crosses below, a sell signal is given. The periods for the fast and slow MAs used in this work were 70 and 140, respectively [10].

VI. RESULTS

TABLE III: Mean MAPE (%)

	AUD	EUR	GBP	NZD	CAD	CHF	JPY
Scaling Law	0.018	0.017	0.017	0.020	0.017	0.018	**0.017**
ML Regression	**0.014**	**0.013**	**0.016**	**0.016**	**0.013**	**0.013**	0.019

TABLE IV: Total Return (%). Values in bold face indicate the best return for a given currency.

	AUD	EUR	GBP	NZD	CAD	CHF	JPY
MLDC	**6.71**	**-0.19**	**10.28**	**8.95**	**16.40**	-3.19	4.84
AvgOS	5.82	-1.86	-5.36	3.37	5.32	-4.77	-0.39
B&H	-8.14	-8.94	-2.41	-9.40	-0.48	**2.75**	**8.61**
2DC	-7.31	-3.03	-5.87	-3.74	-4.86	-4.71	-3.59
TA	-0.72	-2.98	1.96	-3.32	0.64	-4.74	-3.43

TABLE V: Maximum Drawdown (%). Values in bold face indicate the best MDD for a given currency.

	AUD	EUR	GBP	NZD	CAD	CHF	JPY
MLDC	4.32	**2.96**	**2.57**	**2.61**	1.94	3.99	**1.67**
AvgOS	4.73	3.48	5.34	4.77	2.61	5.07	2.69
2DC	7.56	3.16	5.85	4.91	4.94	4.68	3.66
TA	**4.22**	3.02	3.98	4.93	2.53	5.78	3.44

TABLE VI: Calmar Ratio. Values in bold face indicate the best ratio for a given currency.

	AUD	EUR	GBP	NZD	CAD	CHF	JPY
MLDC	**1.55**	**-0.06**	**4.00**	**3.43**	**8.46**	**-0.80**	**2.89**
AvgOS	1.23	-0.53	-1.00	0.71	2.04	-0.94	-0.14
2DC	-0.97	-0.96	-1.00	-0.76	-0.98	-1.01	-0.98
TA	-0.17	-0.99	0.49	-0.67	0.25	-0.82	-1.00

The regression results in Table III show that the ML models perform much better in the regression task than the

TABLE VII: Statistical test results for return (left), maximum drawdown (middle), and Calmar ratio (right), according to the non-parametric Friedman test with the Conover's post-hoc test. Significant differences at the $\alpha = 0.05$ level are shown in boldface. B&H is only included in the returns table, as it only performs a single complete trade (buy on the first day and sell on the last), and as a result maximum drawdown and consequently Calmar ratio cannot be defined.

(a) Returns

Friedman test p-value		**1.53e-22**
	Ave. Rank	p_{Con}
MLDC (c)	1.285714	-
AvgOS	2.857143	0.084397
TA	**3.142857**	**0.043840**
B&H	**3.428571**	**0.021719**
2DC	**4.285714**	**0.002153**

(b) Maximum Drawdown

Friedman test p-value		**1.55e-19**
	Ave. Rank	p_{Con}
MLDC (c)	1.142857	-
TA	2.571429	0.057584
AvgOS	**2.857143**	**0.025572**
2DC	**3.428571**	**0.004491**

(c) Calmar ratio

Friedman test p-value		**2.53e-36**
	Ave. Rank	p_{Con}
MLDC (c)	1.000000	-
AvgOS	2.285714	0.084559
TA	**3.000000**	**0.010869**
2DC	**3.714286**	**0.001163**

benchmark algorithm in all pairs that were tested, apart from USD/JPY. Due to the fact that each currency pair has around 33-75 rolling windows, we present the mean MAPE of all rolling windows for each currency pair. Table III summarises the mean MAPE results across all currency pairs and rolling windows. The mean MAPE for the Scaling Law Regression across all pairs is 0.0180% compared to the mean MAPE of 0.0149% for the ML predictions. This result is also significant as, when tested over all windows using a Kolmogorov-Smirnov test and a null hypothesis stating that both the Scaling Law benchmark and ML regression approaches produce results from the same continuous distribution, a p-value of 3.253e-137 is returned. This significance value is therefore comfortably within a 0.05 significance threshold and the null hypothesis is rejected.

For the trading task we tested the ML based strategy and four benchmark strategies with a fixed transaction cost of 0.0003% applied to each trade. The return results in Table IV shows that the four benchmarks are outperformed by the ML based trading strategy, where nearly all pairs produce a positive return. The significance of these results can be observed in table VIIa showing that the MLDC algorithm ranks first with an average rank of 1.285714, and the Friedman test returns a p-value of 0.009131, demonstrating that there is statistical significance in the total return ranks. With a Conover post-hoc test (p_{Con}), for $\alpha = 0.05$, the MLDC strategy performs significantly better than all benchmarks apart from the AvgOS strategy, which is significant for $\alpha = 0.1$.

The maximum drawdown Table V shows that the MLDC strategy performs better than the benchmark strategies in all but one pair (AUD/USD). The significance of these results can be seen in Table VIIb where the MLDC strategy can be observed to again rank first with an average rank of 1.142857, and significantly outperforming the 2DC and AvgOS strategies for $\alpha = 0.05$. The TA strategy is also significantly outperformed by the MLDC strategy for $\alpha = 0.1$, as it is marginally over the 5% significance level.

Lastly, the Calmar ratio results in Table VI again show that the MLDC strategy outperforms the four benchmarks across all currency pairs. In Table VIIc, MLDC has an average rank of 1.0, and statistically and significantly outperforms the 2DC and TA strategies for $\alpha = 0.05$. The AvgOS strategy however does not meet the significance level of 0.05 with a p-value of 0.084559, so is significant for $\alpha = 0.1$. As the Calmar ratio is an aggregate measure that considers both risk and return, the consistently high ranking of the MLDC strategy is a very

important result, as it demonstrates the algorithm's strength in the risk-return trade off.

Our proposed methodology statistically and significantly outperformed both regression and trading benchmarks. This provides evidence that trading strategies using traditional ML methods to predict the reversal points of DC events can outperform other solely DC-based methods and traditional technical analysis trading methods, as well as the Buy and Hold method.

VII. CONCLUSION

In conclusion, this paper shows that for results obtained over 349 datasets across 7 FX currency pairs: (i) the proposed MLDC algorithm can lead to profitable trading strategies at low risk, (ii) predicting the trend reversal is a key element to a profitable trading strategy, and (iii) using an array of ML algorithms to predict trend reversal is advantageous, as it allows for the best algorithm to be selected for a given dataset. Further work could include different fixed thresholds or a dynamic threshold system.

REFERENCES

[1] D. M. Guillaume, M. M. Dacorogna, R. R. Davé, U. A. Müller, R. B. Olsen, and O. V. Pictet, "From the bird's eye to the microscope: A survey of new stylized facts of the intra-daily foreign exchange markets," *Finance and stochastics*, vol. 1, no. 2, pp. 95–129, 1997.

[2] J. B. Glattfelder, A. Dupuis, and R. B. Olsen, "Patterns in high-frequency fx data: discovery of 12 empirical scaling laws," *Quantitative Finance*, vol. 11, no. 4, pp. 599–614, 2011.

[3] M. Aloud and M. Fasli, "The impact of strategies on the stylized facts in the fx market," Citeseer, Tech. Rep., 2013.

[4] E. P. Tsang, R. Tao, and S. Ma, "Profiling financial market dynamics under directional changes," *Quantitative finance, http://www. tandfon-line. com/doi/abs/10.1080/14697688.2016*, vol. 1164887, 2015.

[5] E. Tsang and J. Chen, "Regime change detection using directional change indicators in the foreign exchange market to chart brexit," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 2, no. 3, pp. 185–193, 2018.

[6] X. Long, M. Kampouridis, and P. Kanellopoulos, "Genetic programming for combining directional changes indicators in international stock markets," in *International Conference on Parallel Problem Solving from Nature*. Springer, 2022, pp. 33–47.

[7] A. Adegboye and M. Kampouridis, "Machine learning classification and regression models for predicting directional changes trend reversal in fx markets," *Expert Systems with Applications*, vol. 173, p. 114645, 2021.

[8] M. Kampouridis and F. E. Otero, "Evolving trading strategies using directional changes," *Expert Systems with Applications*, vol. 73, pp. 145–160, 2017.

[9] O. Salman, M. Kampouridis, and D. Jarchi, "Trading strategies optimization by genetic algorithm under the directional changes paradigm," in *2022 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, 2022, pp. 1–8.

[10] M. Kampouridis, S.-H. Chen, and E. Tsang, "Market fraction hypothesis: A proposed test," *International Review of Financial Analysis*, vol. 23, pp. 41–54, 2012.

Enhancing Solar Panel Efficiency through Deep Deterministic Policy Gradients (DDPG) Reinforcement Learning Control

Diana Ortiz-Munoz
Industrial Engineering and Manufacturing Department
Universidad Autonoma de Ciudad Juarez
Juarez, Mexico
al237849@alumnos.uacj.mx

David Luviano-Cruz
Industrial Engineering and Manufacturing Department
Universidad Autonoma de Ciudad Juarez
Juarez, Mexico
david.luviano@uacj.mx

Luis Perez-Dominguez
Industrial Engineering and Manufacturing Department
Universidad Autonoma de Ciudad Juarez
Juarez, Mexico
luis.dominguez@uacj.mx

Alma Rodriguez-Ramirez
Industrial Engineering and Manufacturing Department
Universidad Autonoma de Ciudad Juarez
Juarez, Mexico
alma.rodriguez.ram@uacj.mx

Abstract— **This study introduces a novel two-degree-of-freedom orientation mechanism for photovoltaic panels, utilizing 3D-printed gears and controlled by the DDPG reinforcement learning algorithm. The research highlights the potential for enhanced solar energy capture. The integration of mechanical design with machine learning showcases a promising interdisciplinary approach to renewable energy systems.**

Keywords—Reinforcement learning, DDPG, photovoltaic panel, orientation control.

I. Introduction (*Heading 1*)

One of the sources of renewable energy that has had a positive development has been solar energy, which is defined as the generation of electrical energy from the reception of solar rays by means of photovoltaic panels [1]. A photovoltaic module (PV) is a device that allows solar energy to be converted into electrical energy without the production of harmful waste for the environment, in addition, they are cheaper and easier to install than wind or biofuel systems [2]. However, the main shortcoming of a PV system is its low efficiency.in the conversion [3]. Therefore, there are 3 ways to improve the efficiency of the conversion of solar energy to electricity: increasing the incidence of solar rays on the panel, improving the conversion efficiency of the electrical diodes, or improving the monitoring of the maximum power point of the panel array.

One way to improve the conversion of solar energy systems is to implement a control system that ensures a greater collection of solar energy with a repositioning of the cell based on the rotation movement of the earth, increasing the daylight hours it receives. This can be achieved with the use of control systems that integrate effective solar tracking algorithms. Recently, reinforcement learning (RL) has been applied to solar tracking systems, providing an adaptive and intelligent control mechanism. RL algorithms learn optimal control policies through trial-and-error, enhancing the system's ability to respond to varying environmental conditions. Models like Q-learning have shown promising results in maximizing solar energy capture [10].

Combining 2-DOF control strategies with reinforcement learning offers a powerful approach to solar tracking. In this paper, we present an innovative mechanical design for the 2-degree-of-freedom orientation of a photovoltaic panel, aimed at maximizing solar energy capture. Our approach leverages the Deep Deterministic Policy Gradients (DDPG) reinforcement learning algorithm, utilizing the intensities of solar rays as system states and controlling angular movements of individual motors as actions [11].

II. Methodology

A. Design of the Orientation Mechanism of the PV

Here we present the design of a two-degree-of-freedom orientation mechanism tailored which offers the flexibility of rotation in two distinct planes, allowing the panel to follow the sun's path both horizontally and vertically. Fig 1 shows the general design of the orientation mechanism.
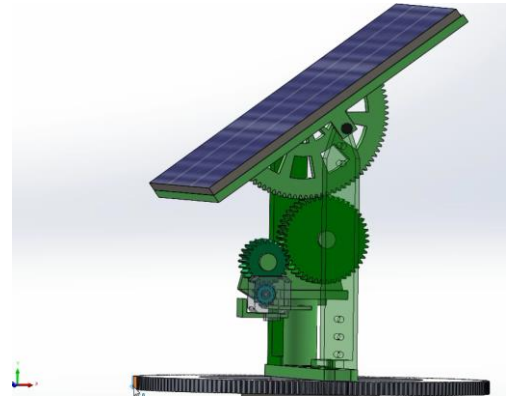

Fig. 1. General arrangement of the design of the orientation mechanism.

B. Angular position control using the DDPG algorithm

With the intention of controlling the angular position of the DC motor in radians, we use the deep reinforcement learning algorithm called DDPG, where a neural network called actor is used to approximate the policy of actions and another neural network called critic la which approximates the state-action functions (Q-values).

The algorithm parameters are shown in Table 1. The reward function used is:

$$\rho = -\left|\theta - \theta_{ref}\right|$$

Where θ is the angular position given in radians. The θ_{ref} value is given by the angle where the greatest amount of intensity of solar rays is received at a given moment by the sensors, at each start of the episode we propose as training start and set-point:

$$\theta_0 = \frac{\pi}{2} * u[0,1] \quad \theta_{ref} = \frac{\pi}{2} * u[0,1]$$

Where $u[0,1]$ is a uniform distribution function in the interval 0,1. So it guarantees us to explore each state and each action within this range of angles.

III. RESULTS OF SIMULATIONS

Fig 2 shows the graph of the reward obtained per episode, the value of the reward converges to a stationary value which results in a policy of deterministic actions.
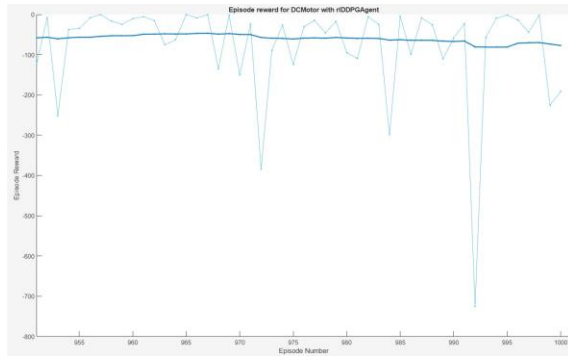

Fig. 2.   . Rewards per episode

TABLE I.   NEURAL NETWORK TRAINING PARAMETERS FOR CRITIC AND ACTOR

Learning Parameter	Value
Layers in the critic's NN	3 layers with 24 neurons
Layers in the actor's NN	5 layers with 24 neurons

Learning Parameter	Value
Actor and critic learning rate	0.001
Sampling time	0.02 seg
buffer length	1000000
discount factor	0.99
lot size	64
Maximum number of episodes	1000
Average number of episodes	50

The results of the training are observed in Table 2

TABLE 2. SIMULATION RESULT

Description	Value
episodes	1000
average reward	-77.09
stop criteria	maximum number of episodes

.
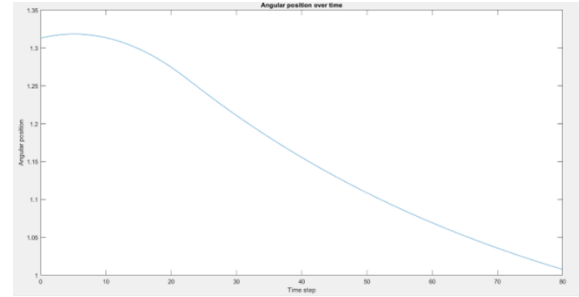Fig 3 shows a tracking from 1.31 rad to 1.03 rad=.

Fig. 3.   Trajectory Tracking, Angular Position of the panel

IV. CONCLUSIONS

The integration of the DDPG algorithm for control demonstrates a promising approach to maximizing solar energy capture. Future work may explore further optimization and real-world applications of this mechanism.

REFERENCES

[1] K. Y. Chou, S. T. Yang, and Y. P. Chen, "Maximum power point tracking of photovoltaic system based on reinforcement learning," Sensors, vol. 19, no. 22, p. 5054, 2019.

[2] R. C. Hsu, C. T. Liu, W. Y. Chen, H. I. Hsieh, and H. L. Wang, "A reinforcement learning-based maximum power point tracking method for photovoltaic array," International Journal of Photoenergy, vol. 2015, 2015.

[3] D. Abel, E. Reif, E. C. Williams, and M. L. Littman, "Toward improving solar panel efficiency using reinforcement learning," in EnviroInfo 2017, 2017.

[4] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, and D. Wierstra, "Continuous control with deep reinforcement learning," arXiv preprint arXiv:1509.02971, 2015

Simultaneous Facial Age Transformation and Reenactment

Jie-Ying Zhang*
Nat'l Taiwan Univ. Sci. Tech.
m11151021@mail.ntust.edu.tw

Li-Syun Hsiung*
Nat'l Taiwan Univ. Sci. Tech.
m11103606@mail.ntust.edu.tw

Gee-Sern Jison Hsu
Nat'l Taiwan Univ. Sci. Tech.
jison@mail.ntust.edu.tw

Abstract—This paper explores concatenating pre-trained models for simultaneous facial age transformation and face reenactment, emphasizing image quality enhancement. We introduce an identity recognition loss function during age transformation model development to separate identity and age features, optimizing it with a finely-tuned age prediction model. Our research highlights the success of this concatenated training process, especially in remarkable image generation results.

Index Terms—age estimation, identity recognition, facial reenactment, age transformation

I. INTRODUCTION

In the realm of computer vision, challenges are prevalent in the domains of facial age transformation and face reenactment. Despite generative model advancements, methods [1] [2] [7] are often specialized, lacking versatility. Even advanced diffusion models struggle with control and identity preservation in facial transformation. Our goal is to create a framework overcoming these challenges, enabling multiple image editing tasks simultaneously. We integrate pre-trained models, coupled with the inclusion of loss functions capable of decorrelating identity and age features to address the challenges of identity preservation. Subsequently, we perform rigorous fine-tuning on the "FFHQ-aging Multi-Pose" dataset to achieve desired outcomes, unlocking new possibilities in this field.

The contributions of this paper are summarized as follows:

- Introduced a training pipeline concatenating pre-trained models for age transformation and face reenactment.
- We propose a data set that involves altering the pose of the original ECAF and SCAF datasets [3], aimed at fine-tuning the age estimator to enhance the quality of the "FFHQ-aging Multi-Pose" data set.

II. RELATED WORK

In age transformation, Hsu et al. [1] balanced age alteration with identity preservation using generative adversarial networks, ensuring effective age transformation. Z. Huang et al. [3] introduced MTLFace, a framework addressing age-invariant face recognition and age synthesis while preserving facial identity.

For face reenactment, Hsu et al. [2] introduced a method that preserves the individual's identity while transferring facial expressions from a source to a target face using a dual-generator architecture.

Our approach combines the Dual-Generator Network [2] and AgeTransGAN [1], enhancing the "FFHQ-aging" dataset [7] with ControlNet [4]. Additionally, we integrate MTLFace to improve AgeTransGAN's capability to accurately preserve facial identity across various age groups.

III. METHODOLOGY

The proposed pipeline comprises three main components: an age estimator for data relabeling, Attention-based Identity Feature Decomposition (AIFD), and the inference stage. Detailed descriptions are provided below.

A. Age Estimator for data relabeling

In the initial phase of our proposal, we employ an Age Estimator to reannotate the Augmented FFHQ-aging dataset. Our Age Estimator implementation closely follows the training process and configuration outlined in [1], demonstrating remarkable performance across various datasets. We initialize the model with pre-trained weights from ImageNet and perform additional pre-training on the IMDB-WIKI dataset, followed by fine-tuning using the alternate SCAF dataset.

B. Attention-based Identity Feature Decomposition (AIFD)

In cross-age face recognition, age disparities lead to variations in facial features, giving rise to distinctive identification challenges. Z. Huang et al. [3] introduced an approach involving the decomposition of combined feature maps within a higher-level semantic domain using an attention mechanism, referred to as "Attention-Based Feature Decomposition" (AFD). The complexity of manipulating feature vectors compared to feature maps underscores the rationale behind this method. Aging or rejuvenation effects, such as the presence of beards and wrinkles, manifest within the semantic feature space but are less evident in one-dimensional features. [3] utilized a ResNet backbone as an encoder to extract mixed feature maps X from an input image I, *i.e.*, $X = E(I)$.

$$X = \underbrace{X \circ \sigma(X)}_{X_{age}} + \underbrace{X \circ (1 - \sigma(X))}_{X_{id}} \tag{1}$$

where $\circ$ indicates element-wise multiplication, and σ is the attention module. The attention module focuses on age-related information in feature maps, guided by an age estimation task, while the residual segment, containing identity related details, is supervised via a face recognition task.
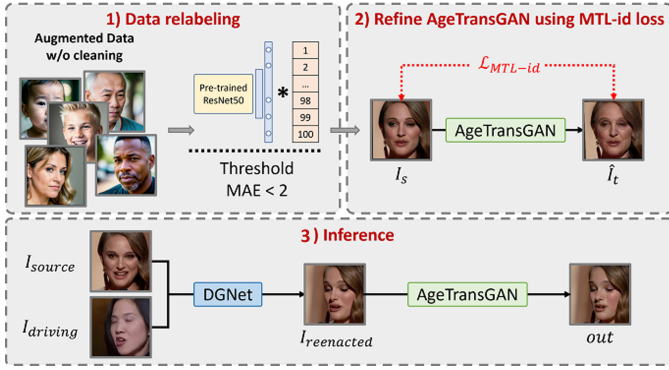
Fig. 1. Outline of the proposed method. The first step covers data preprocessing, the second involves fine-tuning the AgeTransGAN [1] model with MI loss on FFHQ-aging Multi-Pose data set, and the third demonstrates simultaneous facial reenactment and age transformation.

In our approach, we integrate the residual part with identity-related features, passing them through convolutional layers for Age-invariant Face Recognition (AIFR). We introduce an AIFD Encoder, denoted as E_A, for AIFR. Subsequently, E_A encodes input images, creating the multi-task learning id loss (MI loss) function. This loss function integrates into the AgeTransGAN model, enhancing identity feature preservation during fine-tuning on the "FFHQ-aging Multi-Pose" dataset.

$$\mathcal{L}_{MI} = 1 - \cos(E_A(I_s), E_A(\hat{I}_t)) \qquad (2)$$

where E_A is the attention-based feature decomposition (AIFD) Encoder. And I_s stands for the input Image of AgeTransGAN, $\hat{I}_t$ stands for the output.

IV. EXPERIMENT

We employed the "FFHQ-aging Multi-Pose" dataset, previously introduced in prior research [6], containing a total of 10,400 images. Among these images, 90% were designated for the training set, with the remaining 10% reserved for testing.

Given the current scarcity of age datasets featuring multi-angle facial images and the absence of age estimators suitable for evaluating such images, we addressed this gap by augmenting the SCAF [3] and ECAF [3] datasets using the method proposed in [2]. Our augmentation process was designed to incorporate multi-angle facial images into these datasets. Consequently, we designated SCAF as the training set and ECAF as the testing set.

Subsequently, we conducted fine-tuning on the age estimator introduced by [5]. This fine-tuning process resulted in a reduction in the mean age error from 8.54 to 7.22, which enhanced the accuracy of age estimation.

To further improve AgeTransGAN's ability to preserve identity across various age groups, we introduced the MI Loss into the framework. The performance results, including the impact of the MI Loss, are presented in "TABLE I."

In our inference process, as depicted in "Fig.1", we expanded upon previous experiments, successfully achieving simultaneous facial reenactment and age transformation. During inference, we observed a correlation between smaller
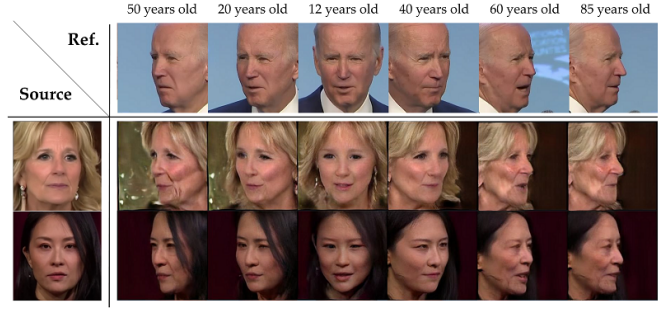

Fig. 2. Inference results for simultaneous face representation and age transformation

TABLE I
COMPARING THE EFFECT OF ADDING MI LOSS TO AGETRANSGAN

Age group	0-2	3-6	7-9	10-14	15-19	30-39	40-49	50-69	70+
Face Verification Rate (%)									
w/o MI Loss	83.2	95.4	98.5	99.3	99.3	99.7	99.3	99.1	93.6
with MI Loss	85.3	96.4	99.1	99.2	99.7	99.6	99.7	99.6	94.2
EAM (Estimated Age Mean)									
Real	1.5	4.9	8.6	12.8	18.9	31.9	43.9	57.2	68.9
w/o MI Loss	2.1	5.2	8.3	13.8	17.6	33.4	41.8	54.6	65.7
with MI Loss	2.1	5.0	8.6	13.4	18.5	34.7	41.2	54.1	65.3

face angles and younger ages, with the correlation gradually increasing as the angles became more acute. These findings are showcased in the high-quality results depicted in "Fig. 2."

V. CONCLUSION

In conclusion, facial manipulation generative models have often been domain-specific, limiting cross-domain transformations. Our approach focuses on concatenating existing models for concurrent facial age transformation and face reenactment. Utilizing the 'FFHQ-aging Multi-Pose' augmented dataset, we optimize it with a finely tuned age transition model and introduce an identity recognition loss function to separate identity from age features. Our research validates the effectiveness of this concatenated training approach, yielding impressive image generation results. In summary, our study advances multi-domain image generation through concatenated models and innovative methodologies with broad applicability.

REFERENCES

[1] Hsu, Gee-Sern and Xie, Rui-Cang and Chen, Zhi-Ting and Lin, Yu-Hong. "AgeTransGAN for Facial Age Transformation with Rectified Performance Metrics." In: ECCV (2022)

[2] Hsu, Gee-Sern, Chun-Hung Tsai, and Hung-Yi Wu. "Dual-generator face reenactment." In: CVPR (2022)

[3] Huang, Zhizhong, Junping Zhang, and Hongming Shan. "When age-invariant face recognition meets face age synthesis: A multi-task learning framework." In: CVPR (2021)

[4] Zhang, Lvmin, and Maneesh Agrawala. "Adding conditional control to text-to-image diffusion models." arXiv:2302.05543 (2023).

[5] Paplham, Jakub, and Vojtech Franc. "Unraveling the Age Estimation Puzzle: Comparative Analysis of Deep Learning Approaches for Facial Age Estimation." In: arXiv:2307.04570 (2023).

[6] Zhang, Jie-Ying, Hsiung, Li-Syun, and Hsu, Gee-Sern. "Unleashing Stable Diffusion for Multi-Modal Facial Image Augmentation.", In: Aris (2023)

[7] Or-El, R., Sengupta, S., Fried, O., Shechtman, E., Kemelmacher-Shlizerman, I. (2020). "Lifespan age transformation synthesis." In: ECCV (2020)

Classification of Songs in Spanish with LLMs: An analysis of the construction of a dataset, through classification

Tania Alcántara
Computational Cognitive Sciences Lab. – CIC
Instituto Politécnico Nacional
México, Ciudad de México
0009-0001-4391-6225

Omar Garcia-Vazquez
Computational Cognitive Sciences Lab. – CIC
Instituto Politécnico Nacional
México, Ciudad de México
omar.gava@hotmail.com

Marco A. Cardoso-Moreno
Computational Cognitive Sciences Lab. – CIC
Instituto Politécnico Nacional
México, Ciudad de México
0009-0001-1072-2985

Hiram Calvo
Computational Cognitive Sciences Lab. – CIC
Instituto Politécnico Nacional
México, Ciudad de México
0000-0003-2836-2102 hcalvo@cic.ipn.mx

Abstract—Songs convey emotions through melody and lyrics. They capture feelings in small text fragments. Emotions within songs vary: positive, negative, or neutral. This study merged two datasets to create a third, leveraging LLMs for competitive song text classification results.

Index Terms—Machine learning, classification, natural language processing, LLMS, Songs

I. Introduction

Emotions are human, arising from biology, culture, and neural interactions. Neurobiology links brain reactions to stimuli, upbringing, and cognition. Music and words evoke emotions. NLP aids emotion classification [1]. Emotion identification in sentences is complex due to subjectivity, culture, and language. Spanish differs across regions. Text length affects accuracy. A project compared general and Mexican Spanish song lyrics, unveiling emotional linguistic subtleties. Merging insights established an emotion classification framework embracing diversity and universal emotions.

II. State of the Art

Efforts in Emotion Classification are notable, particularly in analyzing emotions in Thai songs through polar classification using lyrics [5]. This approach combines a lexicon with traditional machine learning, focusing on emotional features in chorus and verses. [7] employs *SentiWordNet* ontology for extraction, using Naive Bayes, *K-Nearest Neighbor*, and SVM for classification. This method explores *SentiWordNet* ontology's role, emphasizing algorithmic emotion classification in songs. It combines ontology and machine learning to enhance emotional understanding, with *Tweet*-trained models excelling in song analysis, while Google *News* and *Common Crawl* models perform better for movies.

Skip-grams outperform GLoVe, highlighting context's importance. Unconventional model blends like CNN and bi-directional LSTMs achieve 90.66% accuracy [9], enhancing text classification for artistic content.

III. Datasets

The first dataset, **Texts of Songs in Spanish**, is a private collection by the Natural Language Processing Lab at Instituto Politécnico Nacional [10]. It contains 91 Spanish songs from Latin America and Spain, spanning genres like bachata, pop, and ballad. Each song's paragraphs, preserving emotional flow, were divided for coherence, yielding 1,477 items without truncation. "S" stands for neutrality, "P" for positivity, and "N" for negativity, each category a determinate number of examples: S, 97; P, 780; N, 600. The second dataset, **Texts of Songs in Spanish from Mexico**, is from the Computational Cognitive Sciences Lab at Instituto Politécnico Nacional [12]. This set includes 200 Mexican Spanish songs across genres like banda, pop, regional, rock, and cumbia. Each song is meticulously segmented into concise paragraphs, maintaining flow and avoiding incomplete ideas. This process resulted in 4,555 fragments. Labels are S for neutral, P for positive, and N for negative emotions. The nomenclature to identify each category is the same. The dataset 2 has the following distribution: S, 1574; P, 1368; N, 1613.

In previous experiments, the dataset 1 was a better result than in dataset 2, giving the following hypotheses: **Short Phrases:** Dataset two was algorithmically created, with each row representing a sentence from the song. In contrast, dataset 1 was manually compiled, with ideas completed and sentences combined. Merging the two datasets is proposed to maximize their benefits. Alternating the data creates an organic set, ensuring no bias. The datasets have matching columns and

labels, with dataset 2's phrases largely aligned with the new dataset after a 30% length increase. To address accentuated characters in dataset 1's CSV format, UTF-8 encoding was adopted. The new distribution after the union was: S, 2211 (33.11%); P, 2140 (35.90%); N, 1606 (26.95%).

IV. Solution Proposal

A. Preprocessing

For optimal results, preprocess text using conventional techniques: Apply classic data preprocessing – remove, spell check, eliminate stopwords, diagonals, inverted diagonals, numbers, line breaks, parentheses, and double spaces. After initial preprocessing, adapt to Large Scale Language Models (LLM) characteristics. The process consists of two phases. First, models like BERT tokenize text, adding necessary tokens for processing song fragments to capture relationships and contexts. The second phase converts numeric labels to a model-understood format, representing musical traits. Note that LLMs need no extra preprocessing; the aim is accurate context comprehension in song snippets.

B. Classification

It was decided to use the same models as in the first experiments to verify and compare the results. The data set was divided into 80% for training and 20% for validation. The models used for this task were the following:

1) BERT-base-multilingual-cased
2) RoBERTa-base
3) DistilBERT-base-multilingual -cased

Since 2 different data sets were joined, a maximum length of 300 characters was considered, since that was the maximum length used for data set 1.

Table I shows the new configuration for each model:

Hyperparameters	Values		
	RoBERTa	BERT	DistilBERT
Maximum sentence length	300	300	300
Minimum learning rate	0.00004	0.00004	0.00004
Batch size	24	36	48
Evaluation batch size	24	36	48

TABLE I: Modified hyperparameters proposed for the BERT, RoBERTa and DistilBERT

V. Results

The results derived from the datasets and their evaluation through the BERT, RoBERTa and DistilBERT models are presented in table II, making an explicit comparison between each individual dataset, as well as the merged one from our proposal.

It is important to note that a direct comparison with the state of the art is not feasible, since different datasets, preprocessing pipelines and models are involved; therefore, our comparison is directly with our results.

Model	Accuracy Metric		
	Dataset 1	Dataset 2	Dataset 3
RoBERTa	95.66%	91.29%	**96.34%**
BERT	95.58%	91.35%	**96.34%**
DistilBERT	95.83%	92.63%	**95.88%**

TABLE II: Comparative table of the previous experiments and results obtained.

VI. Conclusions and future work

This study focuses on automating categorization of Spanish song texts using models like BERT Multilanguage, RoBERTa, and DistilBERT Multilanguage. These models share similarities in approach. Combining datasets enhances performance, as depicted in Table II.

Consolidating datasets mitigates individual drawbacks, improving overall model performance. Dataset 1 could still be enriched, adding information to enhance language models for better results. In summary, this study illuminates Spanish song text categorization through diverse models, paving the way for refining this technique in future research.

References

[1] Poria, S., Cambria, E., Hazarika, D., Majumder, N., Zadeh, A., & Morency, L. P. (2017). Context-dependent sentiment analysis in user-generated videos. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (pp. 873-883). Association for Computational Linguistics. doi: 10.18653/v1/P17-1081.

[2] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

[3] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692.

[4] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin, I. Attention is all you need. In: Proceedings of the 31st International Conference on Neural Information Processing System, 6000–6010, 2017.

[5] Srinilta, C., Sunhem, W., Tungjitnob, S., & Thasanthiah, S. (2017). Lyric-based Sentiment Polarity Classification of Thai Songs. Proceedings of the International MultiConference of Engineers and Computer Scientists, 1.

[6] Giras, A., Advirkar, A., Patil, C., Khadpe, D., & Pokhare, A. (2014). Lyrics Based Song Genre Classification. Journal of Computing Technologies, 3. Recuperado de http://jctjournals.com/feb2014/v4.pdf.

[7] Kumar, V., & Minz, S. (2013). Mood classification of lyrics using SentiWordNet. En 2013 International Conference on Computer Communication and Informatics (pp. 1-5). doi: 10.1109/ICCCI.2013.6466307.

[8] Çano, E., & Morisio, M. (2020). Quality of Word Embeddings on Sentiment Analysis Tasks. CoRR, abs/2003.03264. Recuperado de https://arxiv.org/abs/2003.03264.

[9] Pham, V. Q., Nguyen, D. T., & Nguyen, T. H. (2020). CNN-BiLSTM model for document-level sentiment analysis. Journal of Information and Telecommunication.

[10] Sidorov, G., (2019) Corpus Textos de Canciones en español. Laboratorio de Procesamiento de Lenguaje Natural, Centro de Investigación en Computación.

[11] Garcia-Vazquez, O, Alcantara, T, Sidorov, G and Calvo H, A. (2023) Análisis de Sentimientos en textos de canciones en idioma español, Research in Computing Science Issue 152(7) (2023)

[12] Alcántara, T., Garcia-Vazquez, O., Calvo, H. and Desiderio, A. (2023) Corpus Textos de Canciones en español mexicano. Laboratorio de Ciencias Cognitivas Computacionales, Centro de Investigación en Computación.

Convolutional autoencoder-based multimodal one-class classification

Firas Laakom*, Fahad Sohrab*, Jenni Raitoharju§ †, Alexandros Iosifidis‡ and Moncef Gabbouj*

*Faculty of Information Technology and Communication Sciences, Tampere University, Finland
§Quality of Information, Finnish Environment Institute, Finland
†Faculty of Information Technology, University of Jyväskylä, Finland
‡Department of Electrical and Computer Engineering, Aarhus University, Denmark

Abstract—**One-class classification refers to approaches of learning using data from a single class only. In this paper, we propose a deep learning one-class classification method suitable for multimodal data, which relies on two convolutional autoencoders jointly trained to reconstruct the positive input data while obtaining the data representations in the latent space as compact as possible. During inference, the distance of the latent representation of an input to the origin can be used as an anomaly score. Experimental results using a multimodal macroinvertebrate image classification dataset show that the proposed multimodal method yields better results as compared to the unimodal approach. Furthermore, study the effect of different input image sizes, and we investigate how recently proposed feature diversity regularizers affect the performance of our approach. We show that such regularizers improve performance.**

Index Terms—**Multimodal learning, one-class classification, anomaly detection, computer vision**

I. PROPOSED APPROACH

In this work, we consider the problem of one-class classification in the presence of multimodal data. We propose an approach that requires multimodal data only from the positive class in the training phase. Here, we describe the formulation for two modalities, while extension to more modalities could be easily obtained. During training, the main target of our approach is to learn a compact mutual embedding of both modalities. Let $\{(x_i, x_i')\}_{i=1}^N$ be the available training data from the positive class, where x_i and x_i' are the first and the second modality of the ith sample, respectively. Our model is composed of two autoencoders, one for each modality. Let $E_1(x_i) \in \mathbb{R}^{m_1 \times p_1 \times d_1}$ be the convolutional output of the encoder on the first modality x_i and $E_2(x_i') \in \mathbb{R}^{m_2 \times p_2 \times d_2}$ be the encoder output of for the second modality. Based on these two outputs, we construct the joint latent representation ϕ_i of the sample (x_i, x_i') as follows:

$$\phi_i = \phi(x_i, x_i') = \text{concat}(\text{Flat}(E_1(x_i)), \text{Flat}(E_2(x_i'))), \quad (1)$$

where $\text{Flat}(\cdot)$ is the flattening operation, i.e, it flattens $E_1(x_i) \in \mathbb{R}^{m_1 \times p_1 \times d_1}$ into a $m_1 p_1 d_1$-dimensional vector. concat is the vector concatenation operation compiling $\phi(x_i, x_i') \in \mathbb{R}^{(m_1 p_1 d_1 + m_2 p_2 d_2)}$ as the final representation of the input.

Our main aim is to learn to map the input data from the positive class into a compact space. This is an objective commonly used by regression-based OCC models and it is usually expressed by minimizing the distance of the latent representations to a pre-defined point [3]–[5]. Using as the target point the origin, the objective becomes to minimize their L_2-norm: $\frac{1}{N} \sum_{i=1}^N ||\phi_i||^2$. By minimizing this loss, the model learns to map the samples from the positive class into a hyper-sphere centered at the origin. In the test phase, any sample that falls close to the origin is assigned to the positive class, and the rest are classified as anomalies.

Minimizing the aforementioned loss can lead to a degenerate solution, i.e., the model learns to map all inputs to the origin and thus fails to distinguish between positive and anomalous samples. To avoid obtaining such solutions, we propose to augment our model using two decoders (one for each modality), aiming to learn to reconstruct the inputs. The outputs of the encoders $E_1(x_i)$ and $E_2(x_i')$ are passed through to the decoders D_1 and D_2. To incorporate the reconstruction objective into the training, we propose to augment the loss in (**??**) using the mean squared loss. The final loss used to train the network can be expressed as follows:

$$L := \frac{1}{N} \sum_{i=1}^N \Big(||\phi_i||^2 + (||D_1(E_1(x_i)) - x_i||^2 +$$
$$||D_2(E_2(x_i')) - x_i'||^2) \Big). \quad (2)$$

The weights of E_1 and E_2 are shared as well as the weights for D_1 and D_2, and they can be trained in an end-to-end manner using gradient-based optimization by minimizing (2). The first term of the loss forces the model to learn a compact representation for both modalities of the same sample in the bottleneck of the architecture, while the other terms regularize the model to avoid degenerate and undesired solutions.

In the test phase, we discard the decoder part. Given a test sample (y, y'), we compile its feature output ϕ, as expressed in (1). The distance of the latent representation of the data point from the origin can be used as an anomaly score. Based on this distance, we can assign the sample to the corresponding class (positive class or negative class). To this end, we need to determine a threshold τ, which is used to define the hyper-sphere enclosing the positive class. The value of τ can be obtained using the training data. As all the training data is

Normal Class	unimodal (left)			unimodal (right)			ours (multimodal)		
	Recall	P@n	ROC	Recall	P@n	ROC	Recall	P@n	ROC
Leptophlebia sp.	0.963	0.921	0.806	0.888	0.897	0.535	0.938	0.921	0.742
Baetis rhodani	0.961	0.893	0.428	0.882	0.892	0.330	0.961	0.907	0.582
Elmis aenea larva	0.844	0.900	0.367	0.896	0.900	0.456	0.935	0.899	0.427
Oulimnius tuberculatus larva	0.896	0.897	0.444	0.909	0.919	0.646	0.896	0.897	0.560
Average	0.916	0.903	0.511	0.894	0.902	0.49	**0.932**	**0.906**	**0.578**

from the positive class, τ can be set to 95th percentile of the feature norms of the training data $\{\phi_i\}_{i=1}^{N}$. Then, given the test sample (y, y'), if $\phi(y, y') \leq \tau$, it is considered to be from the target class. Otherwise, it is considered an anomaly.

It should also be noted that, although in this work, we use shared weights for E_1 and E_2, it is possible to use a different model for every modality. However, one-class classification tasks usually have scarce data [1], [6]. Using shared weights reduces the total number of parameters and acts as a regularization, which makes our model suitable for learning from a limited amount of data.

II. EXPERIMENTS AND DISCUSSION

In this section, we evaluate the performance of the proposed multimodal one-class classification method.

We used a subset of the multimodal image classification dataset of benthic macroinvertebrates, FIN-Benthic [2]. In particular, we used data from 4 classes. Each sample point is presented with two RGB images (which act as the two modalities) from two perpendicular viewpoints. Using this dataset, we constructed four different one-class classification tasks. In each task, data from a single class (out of the four) is considered the normal class, and the remaining three classes are combined to form the anomaly class. In each of the four experiments, we used 66% of normal class data as training data, and we held the rest along with the anomaly data (the remaining three classes) as our test data. All the images were resized to 32×32 pixels.

Our implementation is based on [7]. To train our models, we use Adam optimizer with a learning rate of 0.001 and weight decay of 10^{-3}. The training is conducted with 4 epochs and a batch size of 32. The input image size is 32×32 pixels. For the encoders E_1 and E_2, we used a fully convolutional model which consists of three blocks of convolution, batch normalization, *maxpooling*, and *dropout* layers. All the convolution filters have a size of 3×3 and were selected to be 64, 32, and 16 in the first, second, and third layers, respectively. For the decoder part, i.e., D_1 and D_2, we used the corresponding symmetric layers.

To test the hypothesis that multimodal learning helps in the context of one-class classification, we also experimented with the unimodal variant of the method, i.e., using only one branch of the model and using images from one modality. This yields two competing methods, namely unimodal (left) and unimodal (right), for the left and right modalities, respectively.

In Table I, we report the results for the multimodal model along with the two unimodal models on the four one-class classification tasks. We also report the average results over the four tasks. For each method, we report the Recall scores, the Precision at rank n (P@n), and the Area Under the Curve (ROC) [7].

As shown in Table I, multimodal learning, indeed, yields better performance compared to both unimodal cases in all three metrics. For instance, in the average performance, the proposed multimodal model yields 0.067 and 0.088 improvement in ROC compared to the unimodal models using the left right images, respectively. We also note that on the four tasks, the worst ROC achieved by unimodal (left) and unimodal (right) models are 0.367 and 0.330, respectively, whereas for the multimodal model, the lowest ROC corresponds to *Elmis aenea larva* and is equal to 0.427.

ACKNOWLEDGEMENT

REFERENCES

[1] Fahad Sohrab, Jenni Raitoharju, Moncef Gabbouj, and Alexandros Iosifidis, "Subspace support vector data description," in *2018 24th International Conference on Pattern Recognition (ICPR)*. IEEE, 2018, pp. 722–727.

[2] Jenni Raitoharju, Ekaterina Riabchenko, Iftikhar Ahmad, Alexandros Iosifidis, Moncef Gabbouj, Serkan Kiranyaz, Ville Tirronen, Johanna Ärje, Salme Kärkkäinen, and Kristian Meissner, "Benchmark database for fine-grained image classification of benthic macroinvertebrates," *Image and Vision Computing*, vol. 78, 2018.

[3] Qian Leng, Honggang Qi, Jun Miao, Wentao Zhu, and Guiping Su, "One-class classifcation with extreme learning machine," *Mathematical Problems in Engineering*, vol. 2015, Article ID 412957, 2015.

[4] Alexandros Iosifidis, Vasileios Mygdalis, Anastasios Tefas and Ioannis Pitas, "One-class classification based on extreme learning and geometric class information," *Neural Processing Letters*, vol. 45, no. 2, pp. 577–592, 2017.

[5] Haozhen Dai, Jiuwen Cao, Tianlei Wang, Muqing Deng and Zhixin Yang, "Multilayer one-class extreme learning machine," *Neural Networks*, vol. 115, pp. 11–22, 2009.

[6] Naeem Seliya, Azadeh Abdollah Zadeh, and Taghi M Khoshgoftaar, "A literature review on one-class classification and its potential applications in big data," *Journal of Big Data*, vol. 8, no. 1, pp. 1–31, 2021.

[7] Yue Zhao, Zain Nasrullah, and Zheng Li, "Pyod: A python toolbox for scalable outlier detection," *Journal of Machine Learning Research*, vol. 20, no. 96, 2019.

TransOpt: Transformer-based Representation Learning for Optimization Problem Classification

1st Gjorgjina Cenikj
Computer Systems Department
Jožef Stefan Institute
Jožef Stefan International
Postgraduate School
Ljubljana, Slovenia
gjorgjina.cenikj@ijs.si

2nd Gašper Petelin
Computer Systems Department
Jožef Stefan Institute
Jožef Stefan International
Postgraduate School
Ljubljana, Slovenia
gasper.petelin@ijs.si

3rd Tome Eftimov
Computer Systems Department
Jožef Stefan Institute
Ljubljana, Slovenia
tome.eftimov@ijs.si

Abstract—In this work, we propose a novel representation of optimization problem instances using a transformer-based neural network architecture trained for the task of problem classification of the 24 problem classes from the Black-box Optimization Benchmarking (BBOB) benchmark. We show that transformer-based methods can be trained to recognize problem classes with accuracies in the range of 70%-80% for different problem dimensions, suggesting the possible application of transformer architectures in acquiring representations for black-box optimization problems.

Index Terms—single-objective continuous optimization, representation learning, problem landscape features

I. Introduction

The representation of optimization problems in terms of numerical features, often referred to as problem landscape features, is essential for the Machine Learning based analyses of the similarity and representativeness of problem instances [1]–[3], automated algorithm selection and configuration. Several types of features capturing properties of single-objective continuous optimization problems have been proposed, which can be broadly categorized as explicitly defined features (such as those based on Fitness Landscape Analysis (FLA) [4], Exploratory Landscape Analysis (ELA) [5] and Topological Landscape Analysis (TLA) [6]), and feature-free approaches [7], [8]. However, FLA features require human effort to be computed, while the ELA features can be computationally expensive to compute for high dimensional problems, have been shown to be sensitive to the sample size and sampling method [9], and are not invariant to transformations such as scaling and shifting of the optimization problem [1].

Motivated by the success of transformer-based models [10] in various fields and applications, in this paper, we explore the possibility of generating vectorized representations of optimization problems through the use of transformer models applied to samples of the optimization problem. In particular, we train a transformer-based model architecture for classifying optimization problem instances from the Black-box Optimization Benchmarking (BBOB) [11], [12] suite into one of the 24 problem classes.

II. Methodology

The Black-box Optimization Benchmarking (BBOB) [11], [12] suite contains 24 single-objective optimization problem classes. Each problem class can further have multiple problem instances, which are a transformation of the original problem class. We use the first 999 instances of each problem class to train the transformer model to predict one of the 24 problem classes. We use problem classes of dimensions 3 and 20.

The first step of the proposed methodology involves the generation of samples from which the problem representations are calculated. The samples are obtained using Latin Hypercube Sampling, and we explore sample sizes of $50d$ and $100d$, where d is the problem dimensionality. The objective values of the samples (y-values) are scaled to be in the range [0,1], while the candidate solutions (x-values) are kept in their original range of [-5,5], since this range is fixed in the BBOB.

Figure 1 shows the model architecture. The input to the model are the samples from the optimization problems. A single training instance of the ML model is a matrix of shape $[s, d+1]$, where s is the number of problem instance samples, and d is the dimensionality of the problem. In this case, the second dimension of the input matrix is $d+1$, since the value of the objective function is also included. The encoder part of the transformer model produces an embedding of size e, which is a specified model parameter, for each of the given samples from the optimization problem, i.e. it outputs a matrix of the shape $[s, e]$. In order to obtain a single, flat representation of the problem, we calculate several descriptive statistics on the matrix obtained from the encoder. In particular, we calculate the minimum, maximum, mean, and standard deviation of the representations of the samples produced by the encoder. Concatenating the vectors obtained using each descriptive statistic, we obtain the representation of the problem, which is then fed into a classification head, which produces the class of the problem. The classification head contains a linear layer with a Rectified Linear Unit activation, a dropout layer, and a linear layer which performs the classification into 24 problem classes.

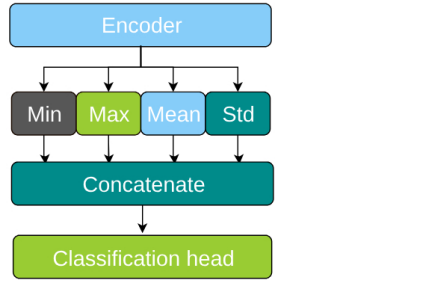The model is trained with a cross-entropy loss, the Adam
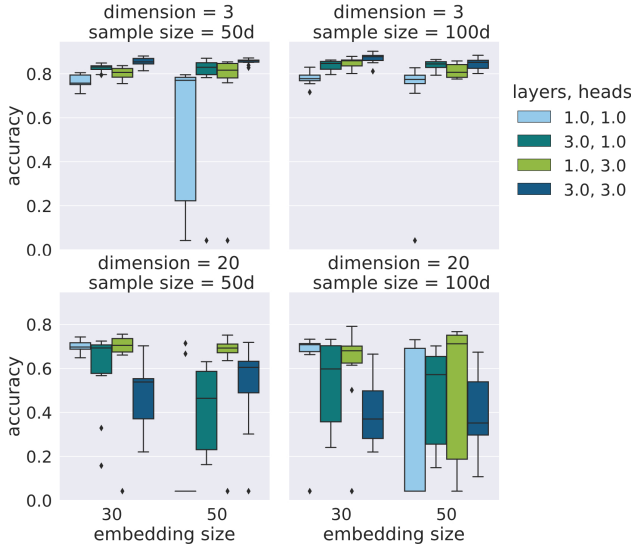
Fig. 1. Model architecture

Fig. 2. Accuracy obtained with different model parameters

models with lower values of the investigated parameters seem to provide satisfactory results for all dimensions. The best performing configuration is (sample size = 50d, embedding size = 30, heads = 1, layers = 1), providing accuracies in the range 70%-80% for problems of different dimensions.

IV. CONCLUSION

We propose a novel representation for single objective optimization problem instances, which uses a transformer model trained for the task of problem classification of samples of the problem instances. The proposed model achieves accuracies around 70%-80% for different problem dimensions, indicating its potential for representing high-dimensional problems and possible utilization as end-to-end algorithm selection models.

V. ACKNOWLEDGEMENTS

REFERENCES

[1] U. Škvorc, T. Eftimov, and P. Korošec, "Understanding the problem space in single-objective numerical optimization using exploratory landscape analysis," *Applied Soft Computing*, vol. 90, p. 106138, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1568494620300788

[2] G. Cenikj, R. Dieter Lang, A. Petrus Engelbrecht, C. Doerr, P. Korošec, and T. Eftimov, "SELECTOR: Selecting a Representative Benchmark Suite for Reproducible Statistical Comparison," in *Proceedings of The Genetic and Evolutionary Computation Conference*, 2022, in Press.

[3] M. A. Muñoz and K. Smith-Miles, "Generating new space-filling test instances for continuous black-box optimization," *Evolutionary Computation*, vol. 28, pp. 379–404, 09 2020.

[4] K. M. Malan and A. P. Engelbrecht, "A survey of techniques for characterising fitness landscapes and some possible ways forward," *Information Sciences*, vol. 241, pp. 148–163, 2013. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0020025513003125

[5] O. Mersmann, B. Bischl, H. Trautmann, M. Preuss, C. Weihs, and G. Rudolph, "Exploratory landscape analysis," in *Proceedings of the 13th annual conference on Genetic and evolutionary computation*, 2011, pp. 829–836.

[6] G. Petelin, G. Cenikj, and T. Eftimov, "Tla: Topological landscape analysis for single-objective continuous optimization problem instances," in *2022 IEEE Symposium Series on Computational Intelligence (SSCI)*, 2022, pp. 1698–1705.

[7] M. V. Seiler, R. P. Prager, P. Kerschke, and H. Trautmann, "A collection of deep learning-based feature-free approaches for characterizing single-objective continuous fitness landscapes," in *Proceedings of the Genetic and Evolutionary Computation Conference*, ser. GECCO '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 657–665. [Online]. Available: https://doi.org/10.1145/3512290.3528834

[8] N. van Stein, F. X. Long, M. Frenzel, P. Krause, M. Gitterle, and T. Bäck, "Doe2vec: Deep-learning based features for exploratory landscape analysis," 03 2023.

[9] Q. Renau, C. Doerr, J. Dreo, and B. Doerr, *Exploratory Landscape Analysis is Strongly Sensitive to the Sampling Strategy*, 09 2020, pp. 139–153.

[10] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2017. [Online]. Available: https://arxiv.org/abs/1706.03762

[11] N. Hansen, S. Finck, R. Ros, and A. Auger, "Real-Parameter Black-Box Optimization Benchmarking 2009: Noiseless Functions Definitions," INRIA, Research Report RR-6829, 2009. [Online]. Available: https://hal.inria.fr/inria-00362633

[12] N. Hansen, A. Auger, R. Ros, O. Mersmann, T. Tušar, and D. Brockhoff, "Coco: A platform for comparing continuous optimizers in a black-box setting," *Optimization Methods and Software*, pp. 1–31, 2020.

optimizer and a maximal learning rate of 0.001. The training is executed for a maximum of 200 epochs, with an early stopping mechanism that prevents overfitting by terminating the training process if the validation loss does not observe a decrease of at least 0.001 for five epochs. We use stratified 10-fold cross-validation to evaluate the model.

III. RESULTS

In order to find a set of reasonable model parameters for further analysis of the architecture, we first ran an initial analysis of the impact of different model parameters on the obtained problem classification accuracy. Figure 2 shows the accuracy obtained in the problem classification task with different parameters for problems of dimension 3 and 20. In particular, we tune the number of heads and layers in the transformer architecture, as well as the dimension of the generated embeddings. The results indicate that increasing the number of heads and layers has a positive impact on the classification accuracy for lower dimensions, however, the same does not always hold for higher dimensions. Increasing the sample size does not seem to substantially impact the classification accuracy for problems of lower dimensions, however, for larger dimensions, the model training is not successful with a sample size of 100d. In general, smaller

Leveraging automation, optimization, and distributed computing to perform high-fidelity regional seismic risk and resilience assessment

Laxman Dahal
Dept. of Civil and Environmental Engineering,
University of California Los Angeles,
Los Angeles, USA.
laxman.dahal@ucla.edu

Henry Burton
Dept. of Civil and Environmental Engineering,
University of California Los Angeles,
Los Angeles, USA.
hvburton@ucla.edu

Kuanshi Zhong
Dept. of Civil and Architectural Engineering and Construction Management,
University of Cincinnati,
Cincinnati, USA.
kuanshi.zhong@uc.edu

ABSTRACT

The ability to quantify regional-scale risk of building inventories is one of the essential components of evaluating community resilience. At its core, resilience-based assessments seek to a quantify the functionality of the buildings, supporting infrastructure, and their interdependencies after an earthquake event. Historically, the mandated design codes for buildings have been governed by life-safety standards with no emphasis on economic loss or functional recovery. Recent advancements in the earthquake engineering domain have put in motion a national-level initiative to make functional recovery the baseline seismic performance standard. While the importance of resilience-based design standards has been broadly acknowledged by engineers, scientists, and policymakers, there is a need to develop efficient regional-scale computational frameworks. Such frameworks would be instrumental in 1) quantifying the risk and resilience of buildings across a region and 2) addressing the computational challenges of performing a large-scale assessment. The quantified regional risk and resilience metrics, in turn, would determine the shape and form in which the community resilience objectives are incorporated into design codes.

The primary objective of this study is to develop a suite of computational engines that leverage automation, optimization, and high-performing computing resources to facilitate high-fidelity (HiFi) seismic risk and resilience assessments. In the context of regional-level assessment, HiFi risk simulations are based on the modern performance-based earthquake engineering (PBEE) methodology, which is designed to conduct individualized and explicit loss (e.g., financial loss and functional recovery time) analysis. The methodology is inherently cumbersome and compute-intensive as it relies on building-, site-, and hazard-specific information that requires substantial data preprocessing. At its core, the methodology systematically transforms seismic hazard into quantifiable risk metrics through three major computational modules: 1) probabilistic seismic hazard analysis (PSHA), 2) probabilistic seismic demand analysis (PSDA), and 3) loss analysis using Monte Carlo simulation. The PSHA is a mathematical representation of the seismic hazard that encompasses uncertainties in the size, location, rate of occurrence, and resulting ground motions that a particular site is likely to observe. Subsequently, the PSDA computes the structural response, ideally via nonlinear response history analyses using the ground motion records selected as part of the PSHA. The distribution of the structural response is ultimately used to perform Monte Carlo simulation-based loss assessment. In this study, the economic loss is assessed following the FEMA P-58 guidelines while the time to regain functionality of a building is calculated based on the ATC-138 procedure. The three modules are executed sequentially as each module hinges on the inputs from the preceding one.

To scale the implementation of a building- and site-specific risk assessment to a region, we developed a Python-based automation engine named "Auto-WoodSDA". Rooted in the PBEE methodology, the Auto-WoodSDA is an end-to-end computational tool that automates code-confirming seismic design, nonlinear dynamic analyses, and loss assessment of an individual building. For the scope of this study, the Auto-WoodSDA is equipped to generate three-dimensional numerical (3D) numerical models of single- and multi-family residential woodframe buildings. Although computationally demanding, the explicit 3D nonlinear time history analysis is adopted as a trade-off to maintain high confidence in the estimated loss metrics. To circumvent the computational inefficiencies at large-scale assessments, we exploit high-performance computing (HPC) resources. The Auto-WoodSDA is a sequential task as seismic design, nonlinear analysis, and loss assessment need to be performed in successive order. However, once the hazard is determined at an individual site, the loss assessment is an independent task that enables the use of distributed computing.

Regional hazard simulation is a complex problem that involves computationally extensive high-dimensional large-scale mathematical simulations. Ultimately, the hazard simulation produces a distribution of shaking intensity maps and representative ground motion records for all the sites across a region. A consistent hazard simulation must consider spatial correlation of shaking intensities both within and between the possible rupture scenarios. The rupture scenarios are selected from the catalog of OpenSHA's Mean UCERF3 model. Moreover, the spatial correlations are incorporated in the intensity maps through eigenvalue decomposition which has a time complexity of $O(n3)$ for n-sites across a region. Once the distribution of shaking intensity is determined, a unique set of ground motion records are selected to enable site-specific nonlinear dynamic analysis. To alleviate the computational burden imposed by hazard simulation and ground motion selection, we leverage the Open MPI library to parallelize the workflow. Specifically, the MPI module is adopted to parallelize the computation on distributed compute nodes in the HPC. The ground motion selection is also parallelized through the MPI module to optimize the Input/Output flow.

In addition to leveraging automation and HPC to address the computation bottleneck, we also optimize the number of unique HiFi simulations such that the computational resources are not exhausted on sites that are likely to experience an identical distribution of losses. Although the optimization routine is trivial, it is based on the proposed strategies that allow the users to control the level of fidelity by defining the granularity of building and hazard representation at a regional scale. The optimization routine is projected to reduce the number of HiFi simulations by 20% in a region with at least 10,000 sites.

Finally, the suite of computational engines is implemented to assess the risk and resilience of residential buildings in the City of Los Angeles. The inventory consists of 17,241 woodframe buildings that were permitted since 2013. First, we performed regional hazard simulation to generate shaking intensity maps and concurrently select 40 ground motion pairs at each site. Utilizing parallel computing across 256 CPUs, the hazard simulation is projected to take 70 CPU hours. Subsequently, an optimization routine is implemented to determine that 14,000 HiFi risk simulations were sufficient to represent the risk at all 17,241 sites. Lastly, Auto-WoodSDA is initialized on 256 distributed CPUs to perform HiFi loss assessment. On average, the end-to-end risk assessment takes 100 minutes per site per node. Assuming 250 nodes are available uninterrupted, the HiFi regional risk simulation is estimated to take less than 250 CPU hours.

Bridging the gap between the HiFi regional risk assessment and computational expense has far-reaching implications. The computational engines developed in this study will pave the path forward to optimize the key building design parameters (e.g., response modification factor (R), importance factor (I_e), and displacement amplification factor (C_d)) to achieve a regional-level functional recovery-based performance objective. Such a study will be crucial in providing a scientific basis for including the resilience-based performance objectives into the design codes.

Context-based classification of sensitive personal information

Sara De Jesús Sánchez
Instituto Politécnico Nacional
Centro de Investigación en
Computación
CDMX, México
sdejesuss2100@alumno.ipn.mx

Eleazar Aguirre Anaya
Instituto Politécnico Nacional
Centro de Investigación en
Computación
CDMX, México
eaguirrea@ipn.mx

Hiram Calvo
Instituto Politécnico Nacional
Centro de Investigación en
Computación
CDMX, México
fcalvo@ipn.mx

Jorge Enrique Coyac Torres
Instituto Politécnico Nacional
Centro de Investigación en
Computación
CDMX, México
jcoyact1900@alumno.ipn.mx

Raúl Acosta Bermejo
Instituto Politécnico Nacional
Centro de Investigación en
Computación
CDMX, México
racostab@ipn.mx

Abstract— **Sensitive personal information is at risk of exposure by the institutions it is shared. Institutions are responsible for preserving the privacy of the personal data they hold, even more so, in the case of sensitive data. ICIS, a model for context-based identification and classification of sensitive personal information, considers the context to identify personal data in unstructured texts of government type documents, regardless the size and type, and then classify each text segment as sensitive personal information, using natural language processing and machine learning techniques. ICIS not only indicates whether a text segment contains sensitive information or not, it also indicates personal data identified in each text segment, their location in the document and whether each text segment is classified as sensitive information. The main contributions of this work are both the identification of personal data and the classification of sensitive information based on the context, and the definition of sensitive personal information, in computational terms.**

Organizations must ensure the confidentiality of the personal information for which they are responsible, according to its level of sensitivity, value, and criticality [1], [2]. Personal data, in general, are those concerning an identified or identifiable natural person [3]. Sensitive personal data are a subset of personal data that refer to the most intimate sphere of its holder, or whose misuse may give rise to discrimination or entail a serious risk to it, such as racial or ethnic origin, present or future state of health, genetic information, biometrics, religious, moral, philosophical beliefs, political opinions, and sexual preferences. Sensitive personal data are subject to specific processing conditions, as they require robust protection [4], [5]. In Mexico, the laws that regulate the processing of personal data carried out by public and private institutions, establish high penalties, and in the case of sensitive personal data, the penalties can be increased up to twice [6]. Many NLP and ML techniques have been used for sensitive personal data classification [7], [8], [9], [10], [11], [12].

ICIS identifies 55 personal data types and classify them as sensitive information, in unstructured texts of government documents, where the writing style is formal and the documents vary in types and sizes. ICIS identifies the different personal data types in each text segment of a document using natural language processing (NLP), then vectorizes the personal data found in each text segment, and classifies each vector as sensitive personal information using machine learning (ML). Both identification and classification are completely related to the context of the speech.

In the identification process, NLP helped to contextualize each type of personal data, which were also identified by the grammatical pattern where they were found. ICIS analyzed not only the word or regular expression match, but also the grammatical pattern in the text segment where the data was found.

For the classification process, ICIS considered the personal context of the text segment. If a text segment contained sensitive data which was not related to a particular person, it was not classified as sensitive information. To be classified as sensitive information, there must be sensitive data related to a particular person. Four personal data classifications were proposed for each text segment, if it contains: identifiers (PID), personal data (PD), sensitive personal data (SPD, and sensitive unit data (SUD). With these data classification results, sensitive information classification rules were proposed to define, in computational terms, what sensitive personal information is (1).

$$SPI = SUD \lor PID \land SPD \tag{1}$$

where:
SPI: Text segment contains sensitive personal information
SUD: Text segment contains sensitive unit data, like Id numbers that may contain sensitive information within their formats.
PID: Text segment contains identifier personal data, like person names.
SPD: Text segment contains sensitive personal data.

Machine learning algorithms performances evaluated for the four personal data classifications were excellent, nB was the best. For the sensitive information classification, the DT algorithm performance was the best.

ICIS can be used as a system component. A path is the input; a json structure with the personal data findings, and the classification results for each text segment in the document, is the output.

Keywords— personal data, sensitive information, machine learning, classification, natural language processing, cybersecurity

REFERENCES

[1] Diario Oficial de la Federación, «Decreto promulgatorio del Protocolo Adicional al Convenio para la Protección de las Personas con respecto al Tratamiento Automatizado de Datos de Carácter Personal, a las Autoridades de Control y a los Flujos Transfronterizos de Datos», 2018. https://www.dof.gob.mx/nota_detalle.php?codigo=5539474&fecha=28/09/2018

[2] OEA, «Clasificacion de datos», OAS, 2019. https://www.oas.org/es/sms/cicte/docs/ESP-Clasificacion-de-Datos.pdf

[3] Comisión Europea, «Grupo de trabajo del articulo 29l Dictamen 4, 2007 sobre el concepto de datos personales. ¿Que son los datos personales?», 2007. https://ec.europa.eu/justice/article-29/documentation/opinion-recommendation/files/2007/wp136_es.pdf

[4] Comisión Europea, «¿Qué datos personales se consideran sensibles?», 2021. https://ec.europa.eu/info/law/law-topic/data-protection/reform/rules-business-and-organisations/legal-grounds-processing-data/sensitive-data/what-personal-data-considered-sensitive_es

[5] UE, «Reglamento (UE) 2019/679 del Parlamento Europeo y del Consejo del 27 de abril de 2016, relativo a la protección de las personas físicas en lo que respecta al tratamiento de datos personales y a la libre circulación de estos datos», 2016. https://eur-lex.europa.eu/legal-content/ES/TXT/?uri=CELEX%3A32016R0679#d1e1547-1-1

[6] INAI 2021, «Ley Federal de Protección de Datos Personales en Posesión de Particulares», 2021. https://home.inai.org.mx/?page_id=1870I\ &mat=p.

[7] Hassan Mathkour, Ameur Touir, Waleed Al-Sanie, «Automatic information classifier using rhetorical structure theory», International Conference on Intelligent Information Processing and Web Mining, 2005.

[8] McDonald, Graham; McDonald, Craig; Ounis, Iadh, «Using Part-of-Speech N-Grams for Sensitive-Text Classification», Proceedings of the 2015 International Conference on The Theory of Information Retrieval, Northampton, Massachusetts, USA, 2015.

[9] Yan Liang, Zepeng Wen, Yizheng Tao, GogLiang Li, Bing Guo, «Automatic Security Classification Based on Incremental Leaning and Similarity Comparison», 2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC), Chongqing, China, 2019.

[10] Guosheng Xu, Chuhao Qi, Hai Yu, Shengwei Xu, Chunlu Zhao, Jing Yuan, «Detecting Sensitive Information of Unstructured Text Using Convolutional Neural Network», 2019 International Conference on Cyber-Enabled Network,» 2019 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), Guilin, China, 2019.

[11] Huimin Jang, Chunling Chen, ShengChen Wu, Yongan Guo, «Classification of Medical Sensitive Data based on Text Classification», 2019 IEEE International Conference on Consumer Electronics - Taiwan (ICCE-TW), Yilan, Taiwan, 2019.

[12] Ji-sung Park, gun-woo Kim, Dong-ho Lee, «Sensitive Data Identification in Structured Data through GenNER Model based on Text Generation and NER», Proceedings of the 2020 International Conference on Computing, Networks and Internet of Things, Sanya, China, 2020.

Profit Allocation in Logistics Enterprise Coalitions Based on Fuzzy Cooperative Game Theory

1st Xi He
Department of Automation
Tsinghua University
Beijing, China
hex21@mails.tsinghua.edu.cn

2nd Shuangxi Huang
Department of Automation
Tsinghua University
Beijing, China
huangsx@tsinghua.edu.cn

Abstract—In the context of e-commerce, establishing a stable alliance is crucial in the logistics industry. Consequently, the challenge of ensuring a fair profit allocation arises. In this study, we address the problem of profit allocation for logistics enterprise coalitions with inadequate information and propose a relevant model. To demonstrate the effectiveness of the model, a case study is provided. The results show that our method enhances the multi-party cooperation and serves as an effective tool for the fair and equitable allocation of profits.

Index Terms—Fuzzy cooperative game theory, Intuitionistic fuzzy set, Logistics enterprise coalition, Profit allocation

1. Introduction

In cooperative games, factors such as cooperation risks and personal preferences often result in partial participation. Aubin introduced the concept of cooperation ratio represented by real numbers in closed interval [1]. Butnariu defined a simplified form of Shapley function for cooperative games with fuzzy coalitions and proportional values [2]. The precise knowledge of achievable payments by each coalition is also often unavailable. Galindo and Gallardo introduced a ranking approach for fuzzy numbers and proposed real Shapley values for games with fuzzy characteristic functions [3]. Liu and Li developed two quadratic programming models to obtain the players' weighted least square contributions [4].

This paper considers a scenario where the participation levels and the profits of the alliance are both uncertain. To address this, we propose several fuzzy Shapley values.

2. Methodology

In this section, some fundamental concepts on fuzzy cooperative games are presented.

2.1 The Uncertainty of Participation Levels

The uncertainty of participation levels is often referred to as the problem of fuzzy coalitions in cooperative game theory. Intuitionistic fuzzy sets have turned out to be a more practical tool to handle the vagueness of participation levels. $\widetilde{S}(i) = \{\langle \mu_s(i), \nu_s(i) \rangle\}$ is an intuitionistic fuzzy set. $0 \leq \mu_s(i) + \nu_s(i) \leq 1$, $\mu_s(i) \in [0,1]$, $\nu_s(i) \in [0,1]$, $\pi_s(i) = 1 - \mu_s(i) - \nu_s(i)$. $\mu_s(i), \nu_s(i), \pi_s(i)$ represent the degree of participation, non-participation and hesitation for partner i to join the alliance $\widetilde{S}$. The intuitionistic fuzzy coalition of the coalition $S \subseteq N$ can be represented as $\widetilde{S}(i) =$ $(\{\langle 1, \mu_s(1), \nu_s(1) \rangle\}, \{\langle 2, \mu_s(2), \nu_s(2) \rangle\}, \ldots, \{\langle n, \mu_s(n), \nu_s(n) \rangle\})$. The whole intuitionistic fuzzy sets are denoted as $IF(N)$.

Assume $[\widetilde{S}_{j_l}] = \{i \in N | \mu(i) \geq j_l\}$, $[\widetilde{S}_{k_m}] = \{i \in N | \omega(i) \geq k_m\}$, $\omega(i) = 1 - \mu(i)$, $j_l, k_m \in [0,1]$, $j_0 = 0, l = 1, 2, \ldots, x(\widetilde{s})$; $k_0 = 0, m = 1, 2, \ldots, x'(\widetilde{s})$. $[\widetilde{S}_{j_l}]$ is the set of partners whose minimum participation level is no less than j_l. $[\widetilde{S}_{k_m}]$ is the set of partners whose maximum participation level is no less than k_m. Let $X(\widetilde{S}) = \{\mu(i) | \mu(i) > 0, i \in N\}$, $X'(\widetilde{S}) = \{\omega(i) | \omega(i) > 0, i \in N\}$, $x(\widetilde{s}), x'(\widetilde{s})$ correspond to the number of elements in $X(\widetilde{S})$, $X'(\widetilde{S})$. By applying the Choquet integral, we can obtain the characteristic function of the intuitionistic fuzzy coalition cooperative game

$$\overline{v}(\widetilde{S}) = [\overline{v}^-(\widetilde{S}), \overline{v}^+(\widetilde{S})] \tag{1}$$

where $v^-(\widetilde{S}) = \sum_{l=1}^{x(\widetilde{s})} v(\widetilde{S}_{j_l})(j_l - j_{l-1})$, $v^+(\widetilde{S}) = \sum_{m=1}^{x'(\widetilde{s})} v(\widetilde{S}_{k_m})(k_m - k_{m-1})$. The pair $\langle IF(N), \overline{v} \rangle$ is a typical form of intuitionistic fuzzy coalition cooperative games.

Given an intuitionistic fuzzy coalition cooperative game, its Shapley value is

$$\overline{\varphi_i} = [\varphi_i^-(\overline{v}), \varphi_i^+(\overline{v})] \tag{2}$$

where $\varphi_i^-(\overline{v}) = \sum_{i=1}^{x(s)} \varphi_i[v(\widetilde{S}_{j_l})](j_l - j_{l-1})$, $\varphi_i^+(\overline{v}) = \sum_{i=1}^{x'(s)} \varphi_i[v(\widetilde{S}_{k_m})](k_m - k_{m-1})$. $\varphi_i[v(\widetilde{S}_{j_l})]$ is the Shapley value of all the participants whose participation level satisfies $\mu(i) \geq j_l$. $\varphi_i[v(\widetilde{S}_{k_m})]$ is the Shapley value of all the participants whose participation level satisfies $\omega(i) \geq k_m$.

2.2 The Uncertainty of Payoffs

The expected profits correspond to each participant's characteristic function. We use the triangular fuzzy numbers to represent coalitions' values as the method depicts the uncertainty of profits in a more applicable way.

A β-cut set of the triangular number $\widetilde{a} = (a^l, a^m, a^r)$ is defined as $\widetilde{a}(\beta) = \{x | \mu_{\widetilde{a}}(x) \geq \beta\}$, where $\beta \in [0,1]$. Thus, a β-cut set of the triangular number $\widetilde{a} = (a^l, a^m, a^r)$ is an interval number, denoted as $\widetilde{a}(\beta) = [a_L(\beta), a_R(\beta)]$ where $a_L(\beta) = (1 - \beta)a^l + \beta a^m$ and $a_R(\beta) = (1 - \beta)a^r + \beta a^m$.

We define an n-person triangular fuzzy cooperative game through an ordered pair $\langle N, \widetilde{v} \rangle$. $N = \{1, 2, \ldots, n\}$ is the set of players and $\widetilde{v}$ is the triangular fuzzy characteristic function of players' coalitions. In general, for any coalition $S(S \subseteq$

N), $\widetilde{v}(S)$ is defined by the triangular fuzzy number $\widetilde{v}(S) = (v^l(S), v^m(S), v^r(S))$, where $v^l(S) \leq v^m(S) \leq v^r(S)$ and $\widetilde{v}(\varnothing) = (0, 0, 0)$.

In order to achieve a fair and balanced profit allocation that maximizes the satisfaction of each participant, we introduce the concept of an excess vector. It measures the satisfaction of partners in the allocation. Given any payoff $x \in R^n$ and participants $i(i \in N)$, an excess vector is defined as $e(i, x) = v(i) - x(i)$ And the distance formula of an α-cut set of the triangular number is given as

$$D(\widetilde{a}(\alpha), \widetilde{b}(\alpha)) = (a_L(\alpha) - b_L(\alpha))^2 + (a_R(\alpha) - b_R(\alpha))^2 \quad (3)$$

Now we can construct a mathematical model to minimize the dissatisfaction of the participants.

$$min\{\sum_{i \in N}[v_L(i)(\alpha) - x_L(i)(\alpha)]^2 + \sum_{i \in N}[v_R(i)(\alpha) - x_R(i)(\alpha)]^2\}$$

$$s.t. \begin{cases} \sum_{i \in N} x_L(i)(\alpha) = v_L(N)(\alpha) \\ \sum_{i \in N} x_R(i)(\alpha) = v_R(N)(\alpha) \end{cases} \quad (4)$$

2.3 Uncertainty of Dual Fuzzy Elements

In this section, we combine the fuzziness of both participation levels and payoffs. Let $N = \{1, 2, \ldots, n\}$ be the set of participants, all the sets of intuitionistic fuzzy sets on N be denoted as $G_{if}(N)$, $\widetilde{v}$ be denoted as a triangular fuzzy number on $G_{if}(N$, which satisfies $\widetilde{v} : G_{if}(N) \to \widetilde{R}$, $\widetilde{v}(\varnothing) = 0$. We define $\langle N, \widetilde{v} \rangle$ as an intuitionistic fuzzy coalition cooperative game whose payoff is a triangular number.

Assume $\widetilde{S} \in G_{if}(N)$. Let $X^-(\widetilde{S}) = \{\mu(i)|\mu(i) > 0, i \in N\}$, $X^+(\widetilde{S}) = \{1 - \nu(i)|1 - \nu(i) > 0, i \in N\}$, $x^-(\widetilde{s})$, $x^+(\widetilde{s})$ correspond to the amount of elements in $X^-(\widetilde{S})$, $X^+(\widetilde{S})$ respectively. Arrange the elements in $X^-(\widetilde{S})$ and $X^+\widetilde{S}$ in monotonically increasing order as $h_1^- < h_2^- < \cdots < h_{x^-(\widetilde{S})}^-$, $h_1^+ < h_2^+ < \cdots < h_{x^+(\widetilde{S})}^+$. Similarly, we can obtain the formula of the Choquet integral of an intuitionistic fuzzy game

$$\gamma^-(\widetilde{S}) = \sum_{i=1}^{x^-(\widetilde{S})} v([\widetilde{S}]_{h_l^-})(h_l^- - h_{l-1}^-) \quad (5)$$

$$\gamma^+(\widetilde{S}) = \sum_{i=1}^{x^+(\widetilde{S})} v([\widetilde{S}]_{h_l^+})(h_l^+ - h_{l-1}^+) \quad (6)$$

Given a confidence level α, v_α be the aggregation form of v at α, we can obtain the Shapley function of $\langle N, v_\alpha \rangle$ as

$$\phi_i(v_\alpha) = \sum_{\substack{i \in \widetilde{S} \\ \widetilde{S} \in G_{if}(N)}} \frac{(|\widetilde{S}| - 1)!(n - \widetilde{S})!}{n!}(v_\alpha(\widetilde{S}) - v_\alpha(\widetilde{S} - \langle i, \mu(i), \nu(i) \rangle)) \quad (7)$$

3. CASE STUDY

One example is set at a county with several logistics enterprises. In rural areas, the delivery of goods is currently affected by factors such as long distances and complex road conditions. Consequently, most logistics companies have not yet established comprehensive coalitions. To gain a higher profit and save more costs, the logistics enterprises adopt the collaborative delivery model. We select three typical logistics companies. Their profits when they operate individually are given in Table 1.

TABLE I
THE INDIVIDUAL PROFITS OF THREE LOGISTICS COMPANIES

Enterprises	Incomes	Costs	Profits
A	[751,2263,3771]	3565	[-2814,-1302,206]
B	[1513,3023,4537]	3565	[-2052,-542,972]
C	[5307,6773,8317]	3565	[1742,3208,4752]
Sum	[7571,12059,16625]	10695	[-3124,1364,5930]

With the coalition formed, the companies can integrate their transportation resource. We apply the method in Section 2.1, 2.2 and 2.3 to calculate the profits allocated to logistics enterprise A, B and C. The results are given in Table 2.

TABLE II
PROFITS ALLOCATED UNDER DIFFERENT STRATEGIES

Strategy	Enterprise A	Enterprise B	Enterprise C
Fuzzy Coaltion	[-804,-412,442]	[-657,12,1573]	[3015,4906,7305]
Fuzzy Payoff	[-633,-353,457]	[-615,57,1615]	[2802,4802,7248]
Dual Elements	[-589,-291,503]	[-561,1644,]	[2704,4695,7173]

The results demonstrate that when logistics enterprises establish coalitions, they are able to achieve higher profits compared to operating in isolation. The overall profit of the local logistics industry can also get enhanced.

4. CONCLUSIONS

In this paper, by combining traditional cooperative game theory with Choquet integral, intuitionistic fuzzy sets and triangular fuzzy numbers, we propose several solutions for fuzzy cooperative games to address more practical problems. The case study suggests that the profits of all the participants shall get increased through the proposed approach.

We will make further research from the following two aspects: (1) Consider more details about the companies and develop a more comprehensive model. (2) When calculating the fuzzy Shapley values, take influencing factors such as risks and innovation into account.

REFERENCES

[1] J.-P. Aubin, "Cooperative Fuzzy Games," *Mathematics of Operations Research*, vol. 6, no. 1, pp. 1–13, 1981.
[2] D. Butnariu, "Stability and Shapley value for an n-persons fuzzy game," *Fuzzy Sets and Systems*, vol. 4, no. 1, pp. 63–72, 1980.
[3] H. Galindo, J. M. Gallardo, and A. Jiménez-Losada, "A real shapley value for cooperative games with fuzzy characteristic function," *Fuzzy Sets and Systems*, vol. 409, pp. 1–14, 2021.
[4] J.-C. Liu and D.-F. Li, "Improved shapley values based on players' least square contributions and their applications in the collaborative profit sharing of the rural e-commerce," *Group Decision and Negotiation*, vol. 31, no. 1, pp. 7–22, 2022.

City Assignment by Multi-Objective Evolutionary Artificial Neural Networks for Multiple TSP

Yoshiaki Katada
dept. Electrical and Electronic Engineering
Setsunan University
Neyagawa, Osaka, Japan
katada @ setsunan.ac.jp

Shinya Watanabe
College of Information and Systems
Muroran Institute of Technology
Mizumoto-cho, Muroran, Japan
sin @ csse.muroran-it.ac.jp

Kazuhiro Ohkura
Graduate School of Advanced Science and
Engineering, Hiroshima University
Higashi-hiroshima, Hiroshima, Japan
kohkura @ hiroshima-u.ac.jp

Abstract—In the multiple traveling salesman problem (TSP), a group of cities to be visited has been assigned to each salesman based only on the cities' geographic information, and the visiting routes of the salesmen are planned. However, there is no guarantee that the adopted clustering method is appropriate for route planning. In this study, we proposed a two-stage search method, where the clustering is performed using an artificial neural network, its weights are designed through a multi-objective evolutionary algorithm (MOEA), and each salesman's visiting route is solved using a TSP solver. We conducted computational experiments for a test problem to compare the performance of the proposed method to a canonical clustering method. Additionally, we examined the characteristics of the balanced solution selected from the obtained non-dominated solution set.

Index Terms—multiple traveling salesman problem, multi-objective evolutionary algorithm, artificial neural networks, clustering

I. Introduction

The TSP is a problem in which geographic information of multiple cities is given, and a salesman needs to find a route that visits all cities exactly once without duplicates and returns to the starting city [1] [2]. While the classic TSP deals with the route problem for a single salesman, the problem of determining the route for multiple salesmen assigned to visit specific cities is called the multiple traveling salesman problem (MTSP).

In MTSP, it has been common to perform assignments based only on cities' geographic information. Then, solving the TSP separately for each assigned city group is often used [3]. However, such an assignment is not guaranteed to reduce the total tour length or improve evaluation criteria specifically designed for applications categorized as TSP.

To address this issue, we propose a two-stage search method [3] using an artificial neural network (ANN) designed through multi-objective evolutionary computation to determine the assignment of the visiting city groups based on multiple evaluation criteria. We then solve the tour of the assigned city groups using an approximation algorithm widely adopted for TSP.

In this study, we conducted computational experiments using a test problem related to TSP to validate the effectiveness of the proposed method compared to a canonical clustering

method. Additionally, we examined the characteristics of the balanced solution selected from the obtained non-dominated solution set.

II. Objective Functions for MTSP

Let L_k represent the tour length of the k-th salesman ($k \in 1, 2, \cdots, K$), and K denotes the total number of available salesmen. In MTSP, we evaluate the tour lengths for all salesmen. This study calculates the average tour length by dividing the sum of tour lengths by the number of assigned salesmen, K', as shown in (2). This function motivates solutions to involve as many available salesmen as possible. Furthermore, we aim to equalize the tour lengths among all salesmen. For this purpose, we adopt a standard deviation of L_k, denoted by (3). Thus, we aim to solve a 2-objective minimization problem.

$$\text{minimize} \quad F(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x})) \quad (1)$$

$$f_1 = \frac{1}{K'} \sum_{k=1}^{K} L_k \quad (2)$$

$$f_2 = \sqrt{\frac{1}{K} \sum_{k=1}^{K} (L_k - \frac{1}{K} \sum_{k=1}^{K} L_k)^2} \quad (3)$$

III. Evolutionary Multi-Objective Optimization

This study used a fully connected feedforward ANN to assign visiting cities. The input consists of normalized city coordinates, and the output represents the salesman k to be assigned. Each neuron's activation function was sigmoid. This setting allows us to partition the city coordinate plane nonlinearly. Furthermore, using MOEA to design the ANN, we can expect an adaptive assignment of visiting cities based on multiple evaluation criteria.

We applied the real-valued evolutionary algorithm MOEA/D-DE [4] to design the weights of the ANN. As the aggregation function for combining multiple single-objective optimization problems, this study used the Tchebycheff function. Here, we normalize f_1 in (2) and f_2 in (3).
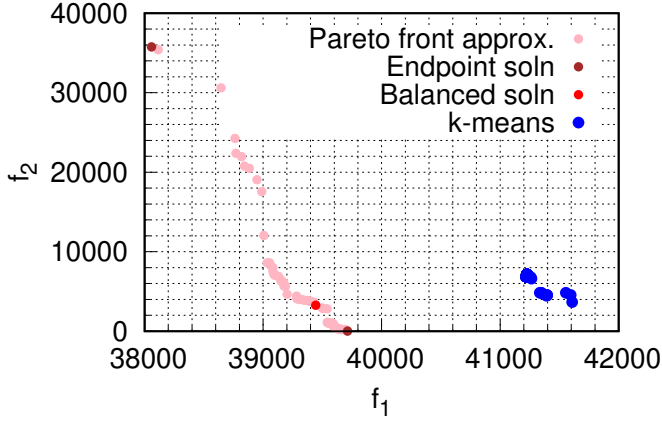
Fig. 1. Objective values for the proposed method and the k-means

(a) balanced solution (inv2) (b) k-means

Fig. 2. Tours obtained for the proposed method and the k-means

TABLE I
THE TOUR LENGTHS AND OBJECTIVE VALUES

	inv1	inv2	inv3	k-means
L_1	47016.8	40350.7	39687.4	34392.3
L_2	1259.0	34471.5	39720.5	54045.1
L_3	43138.8	37345.9	39719.2	40895.8
L_4	97797.1	44100.0	39699.3	40579.7
L_5	1080.5	40946.8	39753.8	36176.9
f_1	38058.4	39443.0	39716.1	41218.0
f_2	35766.9	3282.9	22.6	6885.5

IV. COMPUTER SIMULATIONS

A. Simulation Conditions

This study focused on the fnl4461 from the TSP benchmark. Furthermore, we set the departure city outside the convex hull formed by all city configurations except for the departure city. All salesmen share this city as their common departure city. Thus, the number of cities becomes 4462.

In this problem setting, the number of output nodes N_o of the ANN equals the number of possible salesmen K. In this study, we set the number of input nodes at 2, the number of hidden nodes at 4, and $K = 5$. We used the TSP approximation method LKH-2.0.9 [5] [6] to solve the TSP. We conducted ten trials of the proposed method. We saved all non-dominated solutions obtained during one run as an archive. We also conducted experiments with k-means for city assignment, using the same settings for LKH to compare the solutions obtained by the proposed method.

B. Simulation Results

We show the Pareto front approximation for a typical run of the proposed method and the objective values by the k-means in Fig. 1. The objective values of the solutions obtained by the proposed method have sufficiently converged over those by the k-means. We also present the balanced and endpoint solutions on the approximation as the most representative solutions for the obtained non-dominated solution set.

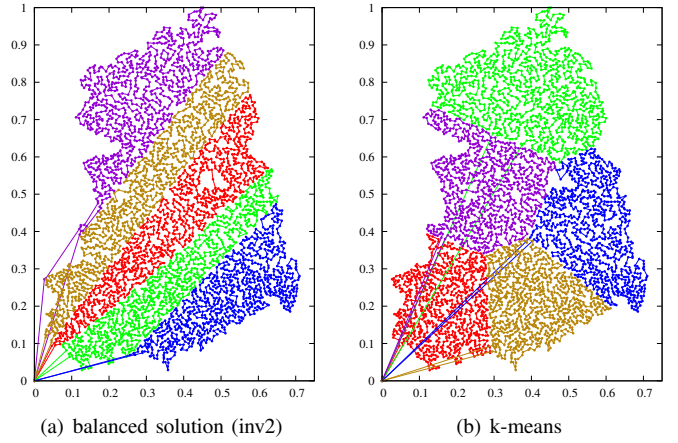The tour lengths and objective values obtained are shown in Table I, where inv1 and inv3 indicate the endpoints and

inv2 indicates the balanced solution. Firstly, inv1 represents a solution with a minimal value of f_1 and a large value of f_2. These values imply that inv1 assigns only a few cities to one or two salesmen, satisfying $K' = K$ while assigning most cities to the remaining salesmen. On the other hand, inv2 represents the balanced solution with the tour lengths uniformed and the small f_1 value. Finally, inv3 has considerable tour lengths and slight deviations among the tour lengths. The objective values of inv2 and inv3 are smaller than those of the k-means.

We show the tours for inv2 and k-means in Fig. 2. In these figures, each color represents the tour of one salesman. The appearance of stripe-like clusters is a characteristic feature of the proposed method observed when the departure city is placed outside all the cities, compared to the k-means.

V. CONCLUSION

In this study, we proposed a method for MTSP using the MOEA and ANN design for city assignment. We conducted computational experiments using a test problem to verify the effectiveness of the proposed method. Additionally, we examined the characteristics of the representative solutions. Consequently, the proposed method's balanced solutions form short tours equalized among each salesman. In the future, we plan to address problems with constraints on the tour lengths of each salesman. Additionally, we aim to apply the proposed method to real-world problems.

REFERENCES

[1] W.J. Cook, In Pursuit of the Traveling Salesman: Mathematics at the Limits of Computation, Princeton University Press, 2012.
[2] Y. Nagata and S. Kobayashi, A Powerful Genetic Algorithm Using Edge Assembly Crossover for the Traveling Salesman Problem, INFORMS Journal on Computing, vol. 25, no. 2, pp. 346–363, 2013.
[3] X. Xu, H. Yuan, M. Liptrott, M. Trovati, Two Phase Heuristic Algorithm for The Multiple-traveling Salesman Problem, Soft Computing, **22**, pp.6567–6581, 2018.
[4] H. Li, Q. Zhang, Multiobjective Optimization Problems With Complicated Pareto Sets, MOEA/D and NSGA-II, IEEE Transactions on Evolutionary Computation, vol. 13, no. 2, pp.284–302, 2009.
[5] K. Helsgaun, General k-opt submoves for the Lin-Kernighan TSP heuristic, *Mathematical Programming Computation*, 1:119–163, 2009.
[6] LKH-2.0.9 http://webhotel4.ruc.dk/ keld/research/LKH/

Optimizing a prediction-based, mixed-asset portfolio including REITs

Fatim Z. Habbab, Michael Kampouridis
School of Computer Science and Electronic Engineering
University of Essex
Wivenhoe Park, United Kingdom
{fh20175, mkampo}@essex.ac.uk

Abstract—The real estate asset class has captured the attention of billions of global investors due to its ability to generate consistent returns and offer diversification benefits within a mixed-asset portfolio. Prior research has highlighted the advantages of including real estate in portfolio optimization. However, existing studies have primarily focused on historical data when addressing this optimization problem. This paper presents an analysis of the performance of a portfolio that incorporates real estate using price predictions derived from a Long Short-Term Memory (LSTM) model. To provide a comprehensive evaluation, we compare the performance of our portfolio against a benchmark portfolio consisting of stocks and bonds only. To this end, we run a genetic algorithm on the two portfolios. Our findings demonstrate a substantial improvement in the average risk-adjusted return of the portfolio that includes real estate with a magnitude of around 100%, highlighting the substantial value that real estate brings to a diversified portfolio. In this way, we propose a novel approach for showing the benefits of investing in real estate.

Index Terms—mixed-asset portfolio, real estate, LSTM, risk-adjusted return

I. INTRODUCTION

Real estate has long been recognized as a valuable component of a diversified investment portfolio [1]. Including real estate in a mixed-asset portfolio offers several potential benefits. First, real estate has historically exhibited low correlation with traditional asset classes such as stocks and bonds [2]. This low correlation can help improve portfolio diversification and reduce overall portfolio risk [3]. Second, real estate investments have the potential to provide consistent income streams through rental income, which can act as a hedge against inflation and provide stability during economic downturns [4]. Third, real estate has the potential for capital appreciation over the long term, as property values tend to increase over time [5].

In order to assess the added value of investing in real estate, it is useful to compare the performance of a portfolio that includes real estate to that of a portfolio that does not include it. Previous studies have examined such comparison and found evidence supporting the inclusion of real estate in mixed-asset portfolio. For instance, [6] found that portfolios that included real estate showed higher risk-adjusted returns compared to portfolios without real estate. In a more recent study, [7] conducted an analysis of U.S. investment portfolios and concluded that portfolios that incorporated real estate assets outperformed those that did not, both in terms of

risk-adjusted returns and diversification benefits. Furthermore, [8] confirmed the positive impact of real estate inclusion on portfolio performance, emphasizing its ability to enhance risk-adjusted returns and improve diversification benefits.

However, those studies relied on historical data in calculating the optimal weights of a mixed-asset portfolio, and then applied those weights to an unseen test set [9]. A potential limitation of such approach is that prices in the test set might differ significantly compared to the prices in the training set [10]. As a result, weights computed using the training set might not fit the test set very well and thus lead to worse portfolio performance (i.e., increased risk and/or reduced return).

To alleviate the above issue, an alternative approach is to try and predict prices in the test set, and then perform the portfolio optimization task — i.e. calculating the optimal weights — directly in the test set [11]. The advantage of this approach is that we focus only on the data period we're interested in — i.e., the test set —; as a result, accurate predictions would closely reflect the prices in the test set, and thus lead to a more efficient portfolio selection. However, the quality of the results is very much dependent on the effectiveness of the price predictions.

Once we have obtained price predictions, we run a genetic algorithm (GA) to optimize a portfolio that includes real estate, stocks and bonds. In order to provide a comprehensive evaluation, we compare the performance of our portfolio to that of a portfolio consisting of stocks and bonds only. Our goal is to demonstrate that a portfolio including real estate outperforms a portfolio not including it. To this end, we evaluate financial metrics such as Sharpe ratio, returns, and risk and compare the results with the proposed benchmark — i.e., portfolio not including real estate.

For our experiments, we used daily prices downloaded from *Yahoo!Finance* for stocks and REITs, and *Investing.com* for bonds, referring to the period between January 2017 and January 2021, for financial instruments belonging to three asset classes — i.e., stocks, bonds, and real estate —, and to three countries — i.e., US, UK, and Australia. For each of the three markets, we used prices for five stocks, five bonds, and five REITs. Thus, we ran our experiments on a total of 90 datasets. All prices were expressed as USD, so as to account for currency risk.

The rest of this paper is organized as follows. Section II explains the methodology used in this study. The results of our experiments are presented in Section III, where we provide a detailed discussion of the results obtained by predicting asset prices using LSTM, and by running a GA to optimize our portfolios. Finally, Section IV summarizes the conclusions of the study.

II. METHODOLOGY

Our experiments aim to provide evidence that a mixed-asset portfolio including real estate can significantly outperform a mixed-asset portfolio not including real estate. This aim can be broken down into two subtasks: (i) use LSTM to predict the prices of REITs, bonds, and stocks, and (ii) use these predictions as an input to a genetic algorithm, which is going to optimize the weights of all assets in the portfolio.

Before applying the LSTM algorithm, we first needed to take several data pre-processing steps, which are presented in Section II-A. We then present the features that we included in the price prediction in Section II-B and the loss function, which is the same across all algorithms, in Section II-C. Afterwards, we briefly present the Python libraries we used to apply our machine learning algorithms in Section II-D. Lastly, we present the genetic algorithm setup in Section II-E.

A. Data preprocessing

Before being used for price prediction, each time series data is differenced and scaled. In this work, we adopt a first-order differencing process to transform our data. First-order differencing is a common technique used in time series analysis to remove the trend component from the data. It aims to transform a non-stationary time series into a stationary one. A stationary time series is one whose statistical properties, such as the mean and variance, remain constant over time. Stationarity is desirable because it simplifies the analysis and makes it easier to model the underlying patterns. By taking the difference between consecutive observations, $D_t = P_t - P_{t-1}$, we obtain a new time series that represents the changes between adjacent data points.

After obtaining the values of D_t, they are further transformed to fall within the range of 0 and 1 using the scaling equation outlined in Equations 1.

$$N_t = \frac{(D - D_{min})}{(D_{max} - D_{min})} \tag{1}$$

where N_t is the standardized value of each variable (in this case the differenced price D), and D_{min} and D_{max} are the minimum and maximum value for D respectively, over all data in each dataset.

B. Features

For our regression problem, we employ two types of features: past observations of a given time series, denoted as N_t, and technical analysis (TA) indicators. The past observations (N_{t-1}, N_{t-2}, N_{t-3}, ..., N_{t-T}) are incorporated as features, with the lag length determined based on the Akaike

Information Criteria (AIC) optimization. AIC is a widely-used metric for model selection. Each dataset may have a different lag length, resulting in a varying number of features. Additionally, we incorporate five TA indicators: Simple Moving Average (SMA), Exponential Moving Average (EMA), Moving Average Convergence/Divergence (MACD), Bollinger Bands, and Momentum. These indicators help identify trends and assist in price prediction and have been shown to improve REITs price prediction accuracy [12]. The SMA represents the average of past prices, while the EMA assigns exponentially decaying weights to past observations. The MACD measures the difference between short-term and long-term EMAs. Bollinger Bands define an interval around the SMA, considering standard deviations from the mean. Momentum captures the difference between prices over a specific time period. These indicators provide valuable information for predicting future price movements.

C. Loss function

Our LSTM model evaluated by using out-of-sample predictions, rather than one-day-ahead predictions. The former is when today's N_t value ($t1$) is known and is used to forecast the value of tomorrow ($t2$). However, tomorrow's value is unknown and cannot be used to forecast the value two days ahead. Hence, this method uses the value forecast at time-step 1 to forecast the value at time-step 2, and so on. In the case of one-day-ahead forecasting, the price today (time-step 0) is known, and is used to forecast tomorrow's price (time-step 1). Then tomorrow's *real* price is used to forecast the price at time-step 2, and so on. This second method is expected to be more accurate, because we are using the actual values as features, instead of predictions. However, for portfolio optimization purposes using out-of-sample predictions would be more realistic as using one-day-ahead predictions would require rebalancing a portfolio on a daily basis for a time period of around 150 days which can lead to significant management costs.

For our problem, we use the *root mean square error* (RMSE) as the loss function, which is presented in Equation 2:

$$\text{RMSE} = \sqrt{\frac{\sum_{t=1}^{|j|}(P_t - \hat{P}_t)^2}{|j|}}, \tag{2}$$

where P_t refers to the actual value of the price, $\hat{P}_t$ is its predicted value, and $|j|$ denotes the number of observations for each dataset j. Please note that as it was explained in Section II-A, the differenced and scaled values (i.e. D_t and N_t respectively) are reverted back to their original price values (i.e. P_t), so that the loss function can be calculated.

D. LSTM

To apply our LSTM algorithm we used the *keras*[1] library. In order to fit the algorithm to the training data we used the

---

[1]https://keras.io/getting_started/ Last access: January 2023

`keras.Sequential` method. The trainable hyperparameters were determined using a grid search method. Once the algorithms were fit to the training data, they were then applied to the test set by using the `predict` attribute of the relevant model.

E. Genetic algorithm

Evolutionary algorithms have been widely used for financial applications — e.g. [13, 14, 15] —, including portfolio optimization [16]. To tackle the portfolio optimization problem we consider in this paper, we use a particular type of evolutionary algorithm known as *genetic algorithm* (GA) [17]. Below we briefly discuss the GA we have used.

GA *chromosomes* (or, *individuals*) consist of N genes indicating the weights allocated to the N assets in the portfolio. The weight are real numbers in the interval $[0, 1]$, and their sum is equal to 1. For example, a GA individual that has the genotype [0.5 0.2 0.3] indicates that there are three assets, and the weight for those asset are 0.5, 0.2, and 0.3, respectively. Initially, all genes are assigned the same weight (in particular, $W_i = 1/N$ for each asset i), which are then evolved according to a set of operators.

We use *elitism*, *one-point crossover* and *one-point mutation*. After the application of crossover and mutation, we apply normalization to each GA individual, to ensure that the sum of weights remains equal to 1.

State-of-the-art methods for solving portfolio optimization problems have used many different metrics as fitness functions. In this paper, we use the *Sharpe ratio*, defined as the ratio of the difference between the average return and the risk-free rate, over the standard deviation of the returns, that is,

$$S = \frac{r - r_f}{\sigma_r}, \tag{3}$$

where r is the average return of the investment, r_f is the risk-free rate, and σ_r is the standard deviation of the returns.

III. RESULTS

In this Section, we examine the experimental results in the form of RMSE distributional statistics (Section III-A), and summary statistics regarding the GA portfolio optimization results (Section III-B). It should be noted that all results are daily results. So when, for example, we present a seemingly "low" return of around 0.03%, its annual equivalent would be around 11.6%. [2]

A. RMSE

First, we compare the accuracy of predictions between two scenarios, one that includes REITs, and one that does not include REITs. Table I shows the summary statistics for two RMSE distributions, one for each of the two previously mentioned scenarios. For each of those distributions, we analyze the mean and standard deviation. As we can observe, the RMSE distribution in the first scenario shows lower RMSE average value compared to the second scenario,

---

[2]AnnualizedReturn $= [(\text{DailyReturn} + 1)^{365} - 1] \times 100 = 11.6\%$.

with a percentage difference of -46.43%. This indicates that including REITs in the analysis improves the accuracy of predictions. Furthermore, the RMSE distribution for the first scenario shows a noticeably lower standard deviation value compared to the second scenario, with a reduction of 50.79%. This suggests that incorporating REITs in the analysis leads to more accurate predictions with reduced variability.

In order to compare the RMSE distributions obtained, we performed a Kolmogorov-Smirnov (KS) test at the 5% significance level. The null hypothesis is that the compared RMSE distributions belong to the same continuous distribution. According to the test results, the adjusted p-value is equal to 1.94E-45, which indicates a statistically significant difference in the two distributions.

In summary, when analyzing the RMSE values, it becomes evident that incorporating REITs in the analysis improves the accuracy of predictions in terms of mean and standard deviation. The scenario of incorporating REITs consistently outperforms the scenario of not including REITs, suggesting that including REITs provides more precise predictions. From the KS test results, we observed that such difference is statistically significant.

B. GA portfolio optimization

After having analyzed the RMSE distributional statistics, we examine the expected portfolio performance for the above-mentioned scenarios. First, we examine the expected return distributions. From Table I, we can notice an increase in the expected return average of around 66.06%. We also notice a 54.11% reduction in the volatility of the expected return distribution, which indicates an increased concentration of values around the mean. This implies that including REITs in a mixed-asset portfolio might improve the overall portfolio return with a reduced volatility.

We also observe that the average expected risk tends to decrease when including REITs with a magnitude of around 33.21%. This implies that investing in REITs allows to reduce the overall portfolio risk. On the other hand, we notice that the standard deviation of the expected risk values tends to decrease with a magnitude of around 63.28%, which indicates an increased concentration of risk values around the mean.

Finally, we observe that the average Sharpe ratio increases when incorporating REITs, with a percentage difference of 103.71%. We also notice a slight increase in the volatility of 4.17%. This suggests that including REITs tends to have a marginal impact on the volatility the risk adjusted returns.

In order to compare the Sharpe ratio distributions obtained, we again performed a Kolmogorov-Smirnov (KS) test at the 5% significance level. Since we are making three comparisons, one for each metric (i.e., portfolio return, risk, and Sharpe ratio), we adjusted the p-values according to the Bonferroni's correction (e.g., $0.05/3 = 0.0167$). According to the test results, the adjusted p-value is equal to 1.55E-45 for all the considered metrics, which indicates a statistically significant difference in the compared distributions.

TABLE I: RMSE and Sharpe ratio distributional statistics. Values in bold represent best results for each statistic.

Metric	RMSE			Expected return		
	Without REITs	With REITs	% Difference	Without REITs	With REITs	% Difference
Mean	36.29	**19.44**	-46.43%	5.41E-04	**8.99E-04**	66.06%
Std Dev	146.15	**71.93**	-50.79%	6.08E-05	**2.79E-05**	-54.11%
Metric	Expected risk			Sharpe ratio		
	Without REITs	With REITs	% Difference	Without REITs	With REITs	% Difference
Mean	5.54E-03	**3.70E-03**	-33.21%	7.26E-03	**1.48E-02**	103.71%
Std Dev	4.04E-04	**1.48E-04**	-63.28%	**5.33E-04**	5.55E-04	4.17%

In summary, when considering the portfolio return, risk, and Sharpe ratio distributions, we observe that including REITs in the analysis has a positive impact on the portfolio performance. It significantly improves the risk-adjusted distributions, as a result of an increased portfolio return and a reduced portfolio risk. The effect of REITs on risk-adjusted return distributions is significant, as shown by the KS test results.

IV. CONCLUSIONS

In our work, we evaluated the performance of a portfolio including REITs by comparing it against a portfolio that does not include REITs. From our experimental results, we noticed a significant improvement in the risk-adjusted performance of our portfolio which is highlighted by a greater average Sharpe ratio that doubles the average Sharpe ratio of a portfolio that does not include REITs. This can be related to a lower average RMSE that results from including REITs in the analysis. This suggests that including REITs in a portfolio including bonds and stocks can mitigate the greater portfolio risk caused by including stock investments.

While our results show that adding real estates to investment portfolios can have positive effect under the diversification perspective, further research can be done on different countries to further explore the opportunities of investing in real estate. Another opportunity for further research might be to extend the holding period for real estate portfolios.

REFERENCES

[1] J. Y. Campbell and L. M. Viceira, "Strategic asset allocation: Portfolio choice for long-term investors," *Oxford University Press*, 2006.

[2] D. M. Geltner, N. G. Miller, J. Clayton, and P. Eichholtz, *Commercial real estate analysis & investments*. Cengage Learning, 2017.

[3] J. Gordon and C. F. Sirmans, *Real estate finance and investments*. McGraw-Hill/Irwin, 2013.

[4] W. B. Brueggeman and J. D. Fisher, *Real estate finance and investments*. McGraw-Hill, 2011.

[5] A. Baum, *Commercial real estate analysis and investments*. Routledge, 2016.

[6] D. C. Ling and A. Naranjo, "Real estate and the capital asset pricing model: The integration of real estate into a mixed-asset portfolio," *Real estate economics*, vol. 30, no. 3, pp. 361–382, 2002.

[7] Z. Chen and D. Ghosh, "Diversification benefits of including commodities, reits, and tips in mixed-asset portfolios: A simulation analysis," *Journal of Real Estate Portfolio Management*, vol. 23, no. 2, pp. 191–201, 2017.

[8] F. Z. Habbab, M. Kampouridis, and A. A. Voudouris, "Optimizing mixed-asset portfolios involving REITs," in *2022 IEEE Symposium on Computational Intelligence for Financial Engineering and Economics (CIFEr)*, pp. 1–8, IEEE, 2022.

[9] J. Lekander and M. Wilhelmsson, "Real estate and the financial crisis: How did the portfolio of swedish real estate firms perform during the crisis?," in *ERES Conference*, 2015.

[10] F. Z. Habbab and M. Kampouridis, "Optimizing mixed-asset portfolios with real estate: why price predictions?," in *2022 IEEE World Congress on Computational Intelligence (WCCI)*, pp. 1–8, IEEE, 2022.

[11] F. Z. Habbab and M. Kampouridis, "Machine Learning for Real Estate Time Series Prediction," in *2022 UK Workshop on Computational Intelligence (UKCI)*, IEEE, 2022.

[12] F. Z. Habbab, M. Kampouridis, and T. Papastylianou, "Improving REITs time series prediction using ML and technical analysis indicators," in *2023 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2023.

[13] S. Cramer, M. Kampouridis, and A. A. Freitas, "Decomposition genetic programming: An extensive evaluation on rainfall prediction in the context of weather derivatives," *Applied Soft Computing*, vol. 70, pp. 208–224, 2018.

[14] M. Kampouridis and F. E. Otero, "Evolving trading strategies using directional changes," *Expert Systems with Applications*, vol. 73, pp. 145–160, 2017.

[15] S. Cramer, M. Kampouridis, A. A. Freitas, and A. Alexandridis, "Stochastic model genetic programming: Deriving pricing equations for rainfall weather derivatives," *Swarm and Evolutionary Computation*, vol. 46, pp. 184–200, 2019.

[16] M. Kampouridis, A. Brabazon, and M. O'Neill, "Applications of genetic programming to finance and economics: past, present, future," *Genetic Programming and Evolvable Machines*, vol. 21 (1), pp. 33–53, 2020.

[17] D. E. Goldberg, "Genetic algorithms in search, optimization, and machine learning. addison," *Reading*, 1989.

Computational Intelligence for Equity-Aware STEM Student Recruitment

Noor Abid and Svetlana Yanushkevich

Biometric Technologies Laboratory, Schulich School of Engineering, University of Calgary, Canada

noor.abid@ucalgary.ca, syanshk@ucalgary.ca

Abstract—**This paper makes a contribution to the CI platform aimed at enhancing the efficiency of student recruitment procedures. Our study entails a comprehensive follow-up audit of this domain, and identifies the key challenges to the integration of equity-conscious practices into the recruitment process. We propose an innovative solution designed to bridge the relevant socio-technological gaps, that is a self-aware recruitment engine. This engine functions within two interconnected conceptual paradigms: machine learning and probabilistic reasoning. To illustrate our approach, we offer a demonstrative example that showcases its practical application.**

Index Terms—**Student recruitment, self-aware engine, STEM, machine learning, causal models, equity, diversity, inclusion**

I. Introduction

Student recruitment is the process of identifying qualified students and persuading them to apply [3]. This process becomes more complicated in the Equity, Diversity, Inclusion (EDI) dimensions [11]. It is influenced by multiple factors such as recruitment strategies, resource availability, demographics, and mindsets. These factors must be harmonized to achieve equity-aware STEM student recruitment, in particular. The fundamental goal of this emerging topic is to re-examine and re-frame the equity-aware STEM student recruitment engine through the lens of computational intelligence (CI). This paper serves as a meaningful contribution to this endeavor.

Student recruitment is considered in various coordinates, such as student-centered quality education, campus, financial considerations, opportunities and resources, and the people at the university [13]. A "student satisfaction" index is a relevant measure that includes university rating, student expectations, perceived quality, perceived value, and student loyalty [12]. Students' interest in STEM careers is influenced by family, out-of-school learning experiences, inside-of-school learning experiences, and media influences [7]. Exploratory and confirmatory factorial analysis used in [15] showed that the predictors of career aspirations include students' opinions on technical topics, the content of school subjects, and related classroom experiences.

From a conceptual standpoint, we distinguish the CI approaches within student recruitment as twofold: a) machine learning, entailing the identification of pertinent data patterns [2], [14], and b) machine reasoning, entailing the provision of expert-guided recommendations for informed decision-making [4], [5].

Limited data on recruitment at STEM universities are available globally. One example is the Norwegian project Lily, aimed at "understanding the priorities, experiences, and motivational factors underlying young people's educational choice" [10]. About 5007 students who chose STEM-related education at a public university or a university college answered a specifically designed survey. In this paper, we use data from [10] for a case study of the causal models combined with EDI-awareness.

II. Self-aware recruitment engine

In this work, we re-examine and re-frame the EDI-aware student recruitment engine based on our previous work [1], using machine reasoning models such as Bayesian Network (BN). Our re-framing calls for the concept of the self-aware learning and reasoning loop illustrated in Fig. 1. The concept of self-awareness has been established in education, psychology, philosophy, and cognitive science [8]. Self-aware computing is a new paradigm for systems to proactively gather information, maintain knowledge about their own internal states and environments, and then utilize this knowledge to reason about behaviors. In Fig. 1, the empirical (perceived) data from an object of observation (a student), is used as a basis for the ongoing learning process. The learned model forms the system's knowledge base, providing the basis for the system's reasoning process. The latter may trigger actions affecting both the behavior of the system (self-adaptation) and possibly impacting the environment.
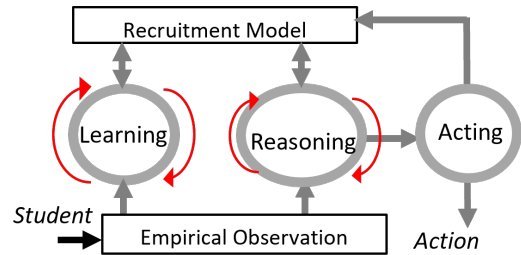

Fig. 1. Concept of self-aware learning-reasoning loop.

Similar to the learning mode of a self-aware recruitment engine, the reasoning mode (Fig. 1) can be implemented using causal networks such as BN, credal networks, Dempster-Shafer networks, and fuzzy causal networks [9]. A BN was chosen in our paper. A causal graph becomes a BN upon assigning its nodes the Conditional Probability Tables (CPTs). Conditional probabilities reflect the probabilistic cause-effect relation.

III. Case Study

In this paper, a subset of the numerical data from [10] was the count of answers to a questionnaire about the person influencing the student's decision to choose STEM. These counts were converted into the frequencies of occurrence, in order to create CPTs as shown in Fig. 2. A fragment of a BN created using pyAgrum library, which is a Python wrapper for the C++ aGrUM library, is shown in Fig. 3. The three nodes in the BN example are Gender, Influencer group, and STEM Choice/Recruitment. There are two values for the variable 'Gender'. The variable 'Influencer group' assumes one of six values: T (Teachers), P (Parents), R (Relatives), F (Friends), A (Acquaintances and Others, and C (Celebrities). The first six columns of the CPT contain '1' to indicate one of the six groups at a time, with other groups encoded as '0'. The last four columns represent the degree (probability) of influence: Minor, Moderate, Medium, or Major.

T	P	R	F	A	C	Minor	Moderate	Medium	Major
1	0	0	0	0	0	0.47	0.24	0.20	0.09
0	1	0	0	0	0	0.20	0.23	0.35	0.22
0	0	1	0	0	0	0.50	0.23	0.18	0.10
0	0	0	1	0	0	0.29	0.28	0.30	0.13
0	0	0	0	1	0	0.34	0.29	0.28	0.10
0	0	0	0	0	1	0.83	0.11	0.04	0.01

Fig. 2. A CPT constructed using the statistics of the degree of influence (from Minor to Major) of the six groups of influencers.
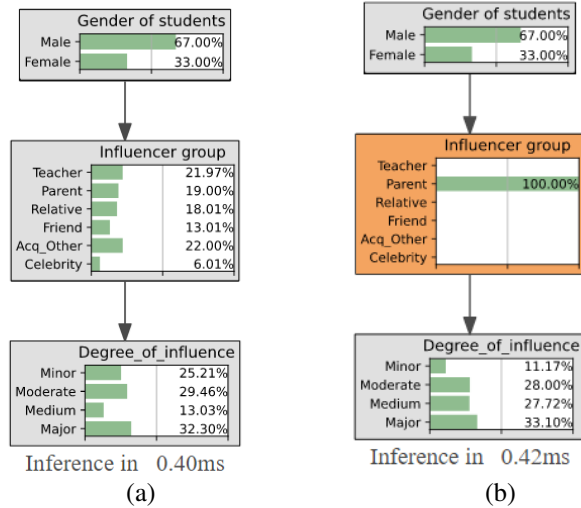

Fig. 3. A causal network and the initial CPTs representing the recruit gender, their indication of the influencers on their decision to choose STEM, and the extent of such influence (a); Inference scenarios that investigate to what degree parents influence the choice of STEM (b).

The simple BN created for this subset of variables has three nodes: the Gender of the first-year undergraduate students participating in the study of their STEM university choice, the Influencer group, and the extent of the Influencer's contribution on the choice of STEM education (Fig. 3).

Consider the following scenario: assuming the influencing group is parents, what is the reported degree of parental influence that students attribute to their decision to pursue STEM? As shown in Fig. 3(b), the Parent group has a Major influence on 33.1% of respondents, followed by Moderate at 28.00%, Medium at 27.72%, while only 11.17% of students described their parents' influence as Minor.

IV. Conclusion and future work

The key conclusions from our work on re-examining and re-framing a recruitment engine for STEM are as follows: (1) There is an imbalance in applying the CI to the task of student recruitment: application of machine learning is dominating while other insightful approaches such as machine reasoning are not given the proper attention. (2) Self-aware computing is useful for exploring achieved results, identifying gaps, systematizing them, and envisioning the horizon. "Dehumanization" of recruitment process identified in [6] addresses the problem of explainability and transparency of CI; we plan to investigate this problem in the EDI context.

References

[1] N. Abid, V. Shmerko, and S. Yanushkevich, Audit of Computational Intelligence Techniques for EDI-aware Systems, *Proc. IEEE Int. Joint Conf. Neural Networks*, Padua, Italy, 2022, pp. 1–8.

[2] A. Bai and S. Hira, An intelligent hybrid deep belief network model for predicting students employability, *Soft Comput.*, 2021, vol. 25, pp. 9241–9254.

[3] G. S. Bapat and S. S. Gankar, Students recruitment strategies at higher educational institutes: A new world perspective – A review of the literature on higher education marketing, *Int. J. Advance Res., Ideas and Innovations in Technology*, 2020, vol. 5, issue 3.

[4] J. A. Chen, M. S. Tutwiler, and J. F Jackson. Mixed-Reality Simulations to Build Capacity for Advocating for Diversity, Equity, and Inclusion in the Geosciences. *J. Diversity in Higher Education*, 2020.

[5] G. Crisp, E. Doran, and N. A. S. Reyes, Predicting Graduation Rates at 4-year Broad Access Institutions Using a Bayesian Modelling Approach, *Res. High Educ.*, 2018, vol.59, pp. 133–155.

[6] M. Fritts and F. Cabrera, AI recruitment algorithms and the dehumanization problem, *Ethics and Inf. Tech.*, 2021, vol. 23, pp. 791–801.

[7] L. Halim, E. Shahali, and Z. Iksan, Effect of environmental factors on students' interest in STEM careers: The mediating role of self-efficacy, *Res. Sci. & Technol. Edu.*, 2021, pp. 1–18.

[8] P. R. Lewis, *et al.*, *Self-aware Computing Systems: An Engineering Approach*, Springer, 2016.

[9] J. Rohmer, Uncertainties in conditional probability tables of discrete Bayesian Belief Networks: A comprehensive review, *Eng. Appl. Artif. Int.*,2020, vol. 88, issue 103384, pp. 1–14.

[10] J. Sjaastad, Sources of Inspiration: The role of significant persons in young people's choice of science in higher education, *Int. J. Sci., Edu.*, 2012, vol. 34, no. 10, pp. 1615–1636.

[11] M. Tamtik and M. Guenter, Policy Analysis of Equity, Diversity and Inclusion Strategies in Canadian Universities – How Far Have We Come?, *Canadian J. Higher Edu.*, 2019, vol. 49, no. 3, pp. 41–56.

[12] A. Turkyilmaz, L. Temizer, and A. Oztekin, A causal analytic approach to student satisfaction index modeling, *Ann. Oper. Res.*, 2018, vol. 263, pp. 565–585.

[13] K. Varner, L. Mey, T. Mentzel, G. Glazer, B. Tobias, and T. Seiple, Pulse Check: Underrepresented Minority Health-Care Student Recruitment and Retention at the University of Cincinnati, *J. College Student Retention: Res., Theory & Practice* 2018, vol. 20, no. 3, pp. 388–405.

[14] P. Verma, S. K. Sood, and S. Kalra, Student career path recommendation in engineering stream based on three-dimensional model, *Comput. Appl En. Educ.*, 2017, vol. 25, pp. 578–593.

[15] M. Virtic and A. Sorgo, Lower secondary school experiences as predictors of career aspirations toward engineering, and production and processing occupations, *Eur. J. Eng. Educ.*, 2022, vol. 47, no. 5, pp. 833–850.

Performance Comparison of Augmented Reality Frameworks

Dafnis Cain Villagran-Vizcarra
dept. de Industria y Manufactura, Instituto de ingeniería y Tecnología
Universidad Autónoma de Ciudad Juárez
Chihuahua, Mexico
al206595@alumnos.uacj.mx

David Luviano-Cruz
dept. de Industria y Manufactura, Instituto de ingeniería y Tecnología
Universidad Autónoma de Ciudad Juárez
Chihuahua, Mexico
david.luviano@uacj.mx

Luis Asunción Pérez-Domínguez
dept. de Industria y Manufactura, Instituto de ingeniería y Tecnología
Universidad Autónoma de Ciudad Juárez
Chihuahua, Mexico
luis.dominguez@uacj.mx

Abstract— **In our quest to incorporate Augmented Reality (AR) into industrial and university laboratories for training purposes, we conducted an analysis of four AR frameworks. Our goal was to develop a portable Starter Kit (SK) and determine the most sustainable option for our project. This kit integrates both hardware and software components, designed to improve an optimize AR functionality on computers, smartphones, and tablets. The research involves four key stages: beginning with NAS configuration, followed by 3D model creation, next is the generation of QR code identifiers, and finally, the development of a cross-platform (C-P) solution.**

Keywords—Augmented reality, augmented reality frameworks, Mobile and Cross-platform

I. INTRODUCTION

AR involves blending digital elements into the physical environment through screens, creating immersive simulations [1]. AR integration into industrial training faces challenges, such as network limitations [2]. This study aims to address these challenges by developing cost-effective AR solutions that can contribute to the wider adoption of AR in the industrial sector.

II. THEORETICAL AND CONCEPTUAL FRAMEWORKS

This section focuses on implementing Augmented Reality (AR) using Android and iOS, with iOS encompassing both iOS and the recent iPadOS [3]. iOS finds versatility in tourist centers and 3D learning environments, with iPads preferred for educational applications [4]. On the other hand, Android, an open-source operating system developed by Google, is widely used in various contexts, including children's reading with AR, video game development, training in reinforced concrete usage, and practice simulators for educational purposes [1]. Vuforia AR, a software development kit (SDK) with Unity, excels in image recognition, text rendering, direction identification, and target tracking, with applications in museums and realistic AR interactions [5]. ARJS, a JavaScript(JS) based framework, simplifies AR development, enhances performance, and accommodates older devices, used in molecular visualization, modeling, and dynamic web content development [6]. Finally, Google's <model-viewer> (MV) is a specialized tag that aids in understanding various components, including AR modeling [7].

III. METHODOLOGY

A. Software and Hardware Configuration

This section covers the essential software and hardware configuration. Android® and iOS® platforms were considered for Augmented Reality (AR) integration, with a requirement of ARCore® certification. Specific mobile phone and tablet models suitable for this project include Motorola, Samsung, Asus, Google, Nokia, and Xiaomi [7].

The selected mobile device is the Xiaomi 10T 5G smartphone, while AR prototypes are developed using a Lenovo Laptop Idea Pad L340 [3]. Unity® is the primary software tool for AR development, with four foundational AR frameworks: ARJS®, EasyAR®, Vuforia®, and Google <Model-Viewer>®.

Vuforia® excels in image recognition and text rendering for applications like museums [5]. EasyAR® is known for its user-friendly interface. ARJS® streamlines development, enables older devices to operate at 60 frames per second, and spans from molecular visualization to dynamic web content [6]. Google <Model-Viewer>® is a proprietary Google framework focused on AR modeling [7].

The Glitch® platform is used for implementing <Model-Viewer> and ARJS® frameworks, and Blender is crucial for creating AR 3D models. Evaluation of AR development across these frameworks involves the participation of an average user for valuable feedback on the user experience.

B. Methodology

The research starts with an AR software literature review for selecting user-friendly AR frameworks, proceeds to a comprehensive analysis of framework webpages for guidance, and then involves creating, testing, and analyzing four prototypes, with a focus on resource utilization.

Active involvement of an average user in tasks such as AR app development and website creation, documented with video recordings, provides valuable insights.

The elements to measure are programming language (L), response time, resources used, additional software, software, application, and platform.

C. Prototype Development Methodology

The prototype development process describes series of steps, with some important requisites of 3D models crafted in Blender, producing file extensions "GBL," "USDZ," "FBX," and "GLTF." The operational course includes distinct phases, each tailored to the unique attributes of the employed AR tools.

1) Google <Model-Viewer> Framework

It begins by visiting the <Model-Viewer> page and selecting the Glitch® hyperlink, leading to preliminary prototype design. "Remix This" is chosen to access code manipulation, and 3D models in "GLB" and "USDZ" formats are added to the "assets" folder.

2) ARJS Framework

"AR.js studio", an open-source AR platform, is employed with a location-based approach, using latitude and longitude coordinates. This processl involves uploading 3D models in "GLB" format and integrating them into an HTML document via the Glitch® platform.

3) EasyAR Framework

EasyAR requires creating an account and downloading the "EasyAR Sense Unity" plugin. Unity® is used with specific components, including "Android® SDK & NDK Tools" and "OpenJDK." The configuration involves company details and image "Target Data."

4) Vuforia Framework

This framework needs a "Vuforia®" account, followed by the download of the "Vuforia Engine SDK and Import Manually" package. The development includes database creation, target data addition, and 3D model integration. This methodology ensures the systematic progression from model generation to prototype development, navigating the distinct attributes of each AR framework with precision.

The methodology consists of seven steps: framework selection, documentation review, prototype development, application or server execution, response time measurement, memory usage monitoring, and data collection. This ensures a systematic progression from model generation to prototype development, tailored to the unique characteristics of each augmented reality framework.

IV. RESULTS

The results of this study are presented in Table 1; however, other findings indicate that Easy AFrame ARJS offers a viable approach to creating applications using HTML files and "glb" format 3D models with image identifiers. Android devices face challenges related to camera permissions, which necessitate the use of Model-Viewer. Model-Viewer streamlines project creation and management on the glitch.com platform, allowing access to "glb" files online. While the Easy AR framework facilitates straightforward prototype development within the Unity environment, the management of 3D modeling files may become complex. Vuforia is another viable option, easy to implement but requiring account verification before utilization.

TABLE. 1. SPECIFICATIONS FOR MEASURING PERFORMANCE, FRAMEWORKS CHARACTERISTICS AND DEVELOPMENT REQUIREMENT.

| | Framework performance | | Frameworks characteristics | | Development requirement |
	Time of answer	Resource used	L	C-P	Application
ARJS	0.7sec	800KB	JS	*	-
MV	1sec	100KB	JS	*	-
Easy-AR	0.2sec	21.5KB	C#	app mobile	23.2MB
Vuforia-AR	0.1sec	26.1KB	C#	app mobile	28.1MB

V. CONCLUSIONS

AR tools streamline development, but evaluating practical features in each framework is crucial. Contemporary AR development is feasible with these tools, fostering research. Identifying user-friendly, adaptable, and fast frameworks is key. These findings highlight AR tools' potential to enhance diverse sectors, driving adoption and innovation. The future of AR promises innovation and progress.

REFERENCES

[1] J. Miguel *et al.*, «La formación de ingenieros en sistemas automotrices mediante la realidad aumentada», *Innovación educativa*, vol. 20, p. 25, jul. 2020.

[2] T. Braud, F. H. Bijarbooneh, D. Chatzopoulos, y P. Hui, «Future Networking Challenges: The Case of Mobile Augmented Reality», *Proceedings - International Conference on Distributed Computing Systems*, n.º June, pp. 1796-1807, 2017, doi: 10.1109/ICDCS.2017.48.

[3] Rocío García, «Qué es iOS: Características y versiones del sistema operativo de Apple». febrero de 2021. Accedido: 30 de mayo de 2021. [En línea]. Disponible en: https://www.adslzone.net/reportajes/software/que-es-ios/

[4] M. Callejas, J. Quiroga, y A. Alarcón, «Ambiente Interactivo Para Visualizar Sitios Turísticos, Mediante Realidad Aumentada Implementando Layar», *Ciencia E Ingeniería Neogranadina*, vol. 21, n.º 2, pp. 91-106, 2011.

[5] Taban, «Introducción a Vuforia (Realidad aumentada) – Taban». mayo de 2018. Accedido: 30 de mayo de 2021. [En línea]. Disponible en: http://taban.mx/2018/05/16/introduccion-a-vuforia-realidad-aumentada/

[6] M. Ramírez, «AR.js acerca la realidad aumentada a la web de forma gratuita | Creativos Online».

[7] Google, «Etiquetas especiales que Google entiende | Centro de la Búsqueda».

Structural Analysis of the Mexico-Toluca Interurban Train with Data Science

Osmar David Arellano-Magdaleno*, Rosa María Valdovinos-Rosas*,
Angélica Guzmán-Ponce†, and David Joaquín Delgado-Hernández *
*Faculty of Engineering, Autonomous University of the Mexico State, Toluca, Mexico
{oarellanom001,rvaldovinosr,david.delgado}@uaemex.mx
† Institute of New Imaging Technologies, Department of Computer Languages and Systems
Universitat Jaume I, 12071 Castelló de la Plana, Spain. aguzman@uji.es

Abstract—As part of an initiative to mitigate transportation congestion of more than 230,000 daily travellers between Mexico City and Mexico State, the Mexican Federal Government, through the Secretariat of Communications and Transportation, began the construction of the Mexico-Toluca Interurban Train in January 2015. In this paper, an exhaustive analysis of the train infrastructure in an event of a high-magnitude earthquake is carried out. For that, a scenario of a large earthquake in which the train moves at high speed transporting passengers is considered. Specifically, we analyse the structure behaviour when it is exposed to an earthquake comparable in magnitude to those experienced on September 19th 1985, in Mexico, and May 22nd 1960, in Chile.

Preliminary results confirm the useful of data science techniques for the study, and offer a comprehensive analysis of the train structural integrity under earthquake conditions via simulations conducted using the SAP 2000 software.

Index Terms—Interurban train, Earthquake, Data Science, Clustering, Machine Learning

I. INTRODUCTION

The goal in this study is to subject the structure of the Mexico-Toluca Interurban Train to a critical load condition before its actual operation, to ensure proper performance once the infrastructure is officially opened.

In this regard, Thai [1] conducts a comprehensive review of Machine Learning applications in structural engineering, structural health monitoring, and concrete structure design. On the other hand, Mangalathu S. and Jeon [2] collected 311 experimental samples of circular columns, considering three types of failures: flexure, flexure-shear and shear; and 6 machine learning methods.

Nguyen et al. [3] introduce a method for damage detection in bridges considering changes in modal curvature using two machine learning models, the Bonghi bridge was used as an illustrative example.

In the same way Mahmoudi et al. [4] introduce a framework for detecting the extent of damages in concrete shear wall buildings during seismic movements using machine learning. The results demonstrate the efficiency of the proposed framework to accurately identify damages in the concrete shear walls of the building.

Abdelijaber et al. [5] introduced an efficient system for real-time detection and localization of structural damages using 1D Convolutional Neural Networks. This approach stands out for its ability to automatically extract optimal features from acceleration signals and its computational efficiency, demonstrated in experiments conducted on a simulator.

In addition, Chow and Ghaboussi [6] proposed a genetic algorithm-based approach for detecting structural damage. Unlike traditional methods, this approach avoids complex analyses and can accurately determine the approximate location of the damage using few measures.

In the Sarothi et al. [7] study, the failure modes in double shear bolted connections in structural steel were explored. For that, different algorithms on a database of 455 experimental results to identify the failure modes were used.

Given the empirical evidence from the studies mentioned above, to evaluate the potential response of the Mexico-Toluca Interurban Train under the circumstances that involve simultaneously a high-magnitude earthquakes and excessive load conditions is imperative. This analysis can offer assessment of the train reliability and the prospective users safety.

II. PRELIMINARIES

The main contribution of this paper is presented from two perspectives:

1) Descriptive Phase: The aim is to analyze the data quality. The original data are distributed into four categories related to the seismic amplification factor. By applying clustering algorithms, its anticipate determining the purity of the data regarding its original assignment to each class. This consideration is particularly significant because the data were synthetically obtained. The k-means and Expectation-Maximization (EM) algorithms were used in this analysis.

2) Predictive Phase: Once the data clustering has been identified, the prediction achievable will be analyzed using three state-of-the-art Machine Learning models: Multilayer Perceptron, J48 decision tree, and Support Vector Machine.

A. Data Understanding

The structural analysis incorporates two specific earthquakes: one that took place on May 22nd 1960, in Chile and another occurred on September 19th 1985, in Mexico (magnitude of 8.1). The Chilean earthquake is significant in

this analysis, being the highest earthquake recorded in global seismic history, with a moment magnitude of 9.5 Mw. The Mexican earthquake was considered due to its proximity to the construction site and status as one of the highest magnitude events in recent years (8.1 Mw).

Based on the reference values related to the physical structure of the Mexico-Toluca Interurban Train, a structural model was formulated [8]. From this model and employing the Structural Analysis Program (SAP 2000) several simulations were conducted: 996 for the Chilean earthquake and 10,000 for the Mexican one (Figure 1).
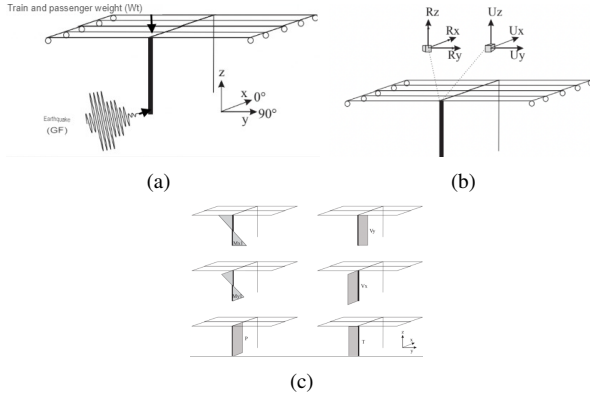
Fig. 1. Overview of key variables applied in designing and assessing a "typical" support pile for the elevated viaduct of the Mexico-Toluca train. (a) Earthquake considerations: the total weight of the train and the weight of the passengers, (b) Variables concerning displacements at the top node, and (c) Variables associated with force exertions.

III. Revelations in Research

For this study, we use the 10-fold cross-validation, being 80% of the instances for training and the remaining 20% for testing. The performance of the models was evaluated using two standard measures obtained from a confusion matrix: accuracy and geometric mean. Table I shows the average values for overall accuracy and the geometric mean from the three Machine Learning models used. The best results are highlighted in bold.

TABLE I
COMPARATIVE RESULTS OF EARTHQUAKE DATASET CLASSIFICATION.

CD	Accuracy		Geometric Mean	
	Chile	Mexico	Chile	Mexico
Multilayer Perceptron				
Original	80.8	40.68	85.1	57.2
EM	97.8	**99.99**	**98.3**	**99.9**
K-Means	**99.3**	99.50	99.1	99.5
Support Vector Machine				
Original	76.9	40.74	82.0	57.7
EM	97.6	**99.99**	98.1	**99.9**
K-Means	**99.0**	99.62	**98.8**	99.6
Decision Tree J48				
Original	80.2	37.33	86.6	54.6
EM	98.3	**99.99**	**98.7**	**99.9**
K-Means	**98.7**	99.12	98.5	99.1

Based on results shown in the Table I, we can observe that: the use of any dataset generated by clustering methods signif-

icantly improves classification performance. When analyzing the accuracy and the geometric mean, both measures display minimal discrepancies between them. This last suggests the class imbalance problem does not affect significantly the performance measures.

IV. Concluding Remarks

This study examines the behaviour that the structure of interurban train might have in earthquakes with magnitudes of either 9.5 or 8.1.

Preliminary results suggest a class overlap problem in the original data. As a result, the performance derived from the clustering algorithms challenges the distribution obtained with the SAP 2000 software. Because it overlap influences the performance of models dificulting the correct recognition between classes.

On the other hand, in terms of the classification, considering both the initially defined classes and the determined through clustering algorithms, the results of the classifier were better when they are trained with datasets clustered. In fact, the clustering obtained by EM gives a superior performance.

Open lines include to cleaning the decision boundering to handle the class overlap issue, and to use a synthetic data generation conserving the statistical distributions as approach to augment data volume and, in this way, increase the performance of the classifiers. From the civil engineering viewpoint, further research will determine the reliability of the structure.

References

[1] H.-T. Thai, "Machine learning for structural engineering: A state-of-the-art review," *Structures*, vol. 38, pp. 448–491, 2022.

[2] S. Mangalathu and J.-S. Jeon, "Machine learning–based failure mode recognition of circular reinforced concrete bridge columns: Comparative study," *Journal of Structural Engineering*, vol. 145, no. 10, p. 04019104, 2019.

[3] D. H. Nguyen, Q. B. Nguyen, T. Bui-Tien, G. De Roeck, and M. Abdel Wahab, "Damage detection in girder bridges using modal curvatures gapped smoothing method and convolutional neural network: Application to bo nghi bridge," *Theoretical and Applied Fracture Mechanics*, vol. 109, p. 102728, 2020.

[4] H. Mahmoudi, M. Bitaraf, M. Salkhordeh, and S. Soroushian, "A rapid machine learning-based damage detection algorithm for identifying the extent of damage in concrete shear-wall buildings," *Structures*, vol. 47, pp. 482–499, 2023.

[5] O. Abdeljaber, O. Avci, S. Kiranyaz, M. Gabbouj, and D. J. Inman, "Real-time vibration-based structural damage detection using one-dimensional convolutional neural networks," *Journal of Sound and Vibration*, vol. 388, pp. 154–170, 2017.

[6] J.-H. Chou and J. Ghaboussi, "Genetic algorithm in structural damage detection," *Computers & Structures*, vol. 79, no. 14, pp. 1335–1353, 2001.

[7] S. Zakir Sarothi, K. Sakil Ahmed, N. Imtiaz Khan, A. Ahmed, and M. L. Nehdi, "Machine learning-based failure mode identification of double shear bolted connections in structural steel," *Engineering Failure Analysis*, vol. 139, p. 106471, 2022.

[8] J. A. Jímenez García, "Función del enlace técnico dentro de la obra del tren Interurbano México Toluca." Memoria de Experiencia Laboral, Ingeniería Civil, Facultad de Ingeniería, Universidad Autónoma del Estado de México, 2017.

Neural Network Regression for Structural Health Monitoring Using Smartphones

Yingqin Zhu
Departamento de Control Automático
CINVESTAV-IPN
Mexico City, 0736, Mexico

Xiaoou Li
Departamento de Computación
CINVESTAV-IPN
Mexico City, 0736, Mexico

Brisbane Ovilla-Martinez
Departamento de Computación
CINVESTAV-IPN
Mexico City, 0736, Mexico

I. Introduction

Structural health monitoring (SHM) plays a vital role in ensuring the safety and durability of aerospace, mechanical, and civil engineering structures. SHM refers to the process of acquiring, validating, and analyzing technical data to aid in life cycle management decisions for a structure. It involves the development of a system capable of detecting changes in a structure caused by damage or deterioration during normal operation, as well as unexpected environmental conditions like earthquakes. A structure's behavior, such as its vibration response, can be influenced by a variety of factors, including sudden or gradual changes in its states, loading conditions, and response mechanisms

In this paper, we have developed a smartphone-based SHM platform that provides an alternative, easier-to-implement method at a lower cost compared to other existing approaches. We propose the utilization of an algorithm based on echo state neural networks, which models the damage in buildings using data obtained from smartphones placed in different parts of the structure. To validate our methodology, we have designed an experiment that employs a dedicated test station for testing the effectiveness of structural damage detection algorithms. Furthermore, we have conducted a comparative analysis of this methodology against classical methods.

II. Smartphone-Based Structural Health Monitoring

Smartphones are ideal for Structural Health Monitoring (SHM) due to their built-in sensors (such as accelerometers), advanced operating systems (like iOS and Android), affordability, and mobility for easy data collection. Communication is carried out through the TCP/IP protocol, with the client receiving command sequences from the host smartphone via a unique IP address. The smartphones and computer connect to the network through a WiFi router. Our SHM system is wireless and works locally. We use Android and IOS as our smartphone operating system because it is partly open source and has libraries that allow direct access to internal peripheral devices, like the accelerometer and gyroscope. Data from these devices is read via a sensor management service at regular intervals. Our proposed smartphone based SHM system is shwon in Fig.1. The smartphones used in our proposed SHM system have accelerometers capable of measuring linear acceleration along the X, Y, and Z axes. We use a computer with Windows 10 operating system to manage communication between all SHM components, store and analyze building data, and run the damage detection algorithm, see Fig.2.
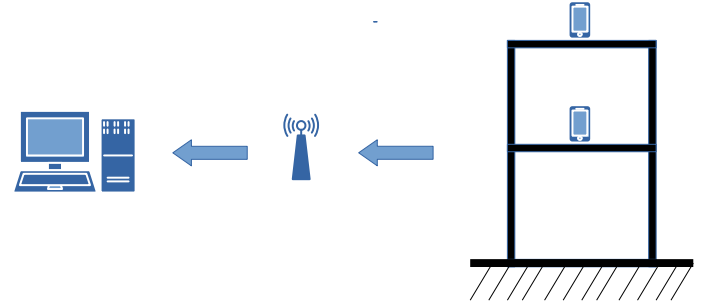

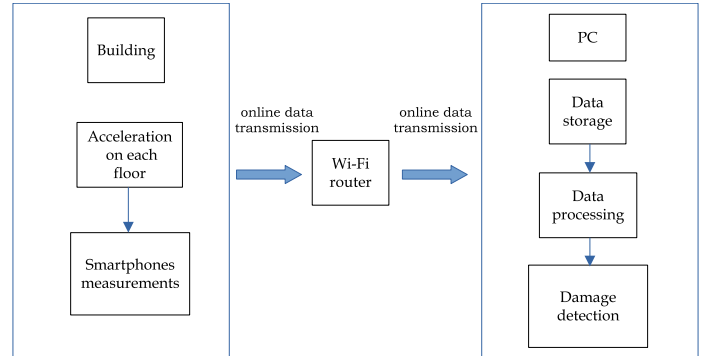Fig. 1. Scheme of smartphone based SHM system.

Fig. 2. The operation of the smartphone based SHM system.

III. Neural network regression for SHM

Prior to applying our approach to identify damage in the testing station, we must first process the data obtained from the smartphones by employing a signal reconstruction technique and a filter to mitigate any noise present in the measurements. Next, we will employ the ground-based data as the input to the network and utilize the damaged or undamaged data as the target for training. The output weights obtained from the training will subsequently be utilized for structural health monitoring as shown in Fig. 4.
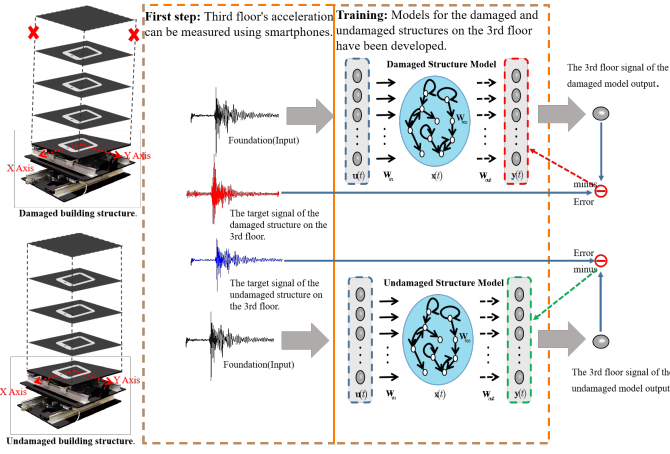
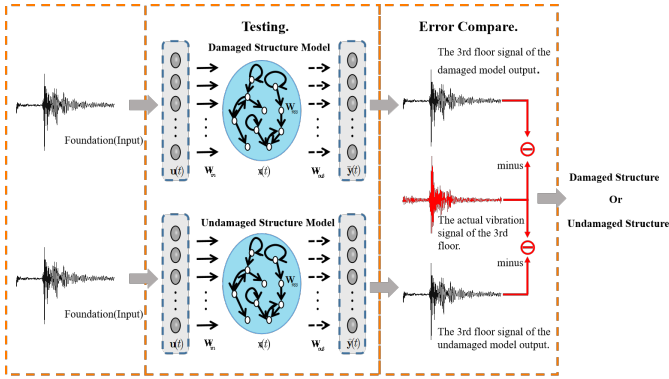Fig. 3. Proposed smartphone based SHM simulation system.

Fig. 4. Detection Phase of SHM System.

$$y(t) = Wx(t) \tag{1}$$

$$W = Y^{target} X^T (XX^T + \beta I)^{-1} \tag{2}$$

IV. EXPERIMENTAL RESULTS

V. CONCLUSION

An SHM system based on smartphones was proposed in this paper, taking advantage of their mobility and ability to communicate through WiFi networks. The system utilized sensors within the phones to measure linear acceleration on each floor of a building, which was then analyzed and processed using a neural network for damage detection. The use of smartphones and NN allows this methodology to be applied to larger structures with multiple floors and other physical structures, due to their communication processes and data processing capabilities.
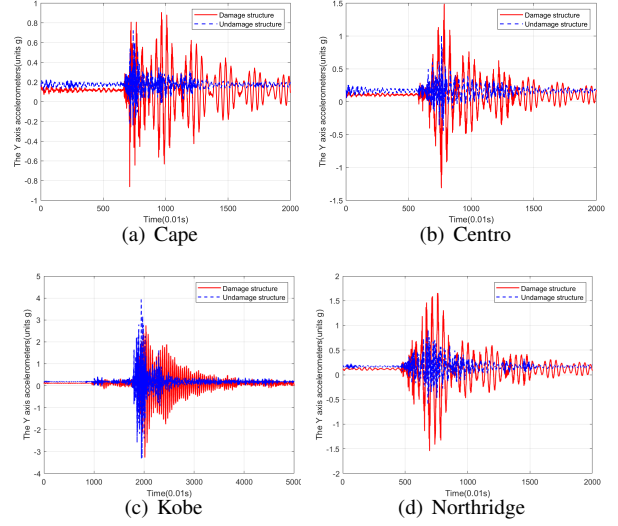

(a) Cape
(b) Centro
(c) Kobe
(d) Northridge

Fig. 5. Comparison of acceleration data along the Y-axis for both undamaged and damaged structures during four types of earthquake simulations.
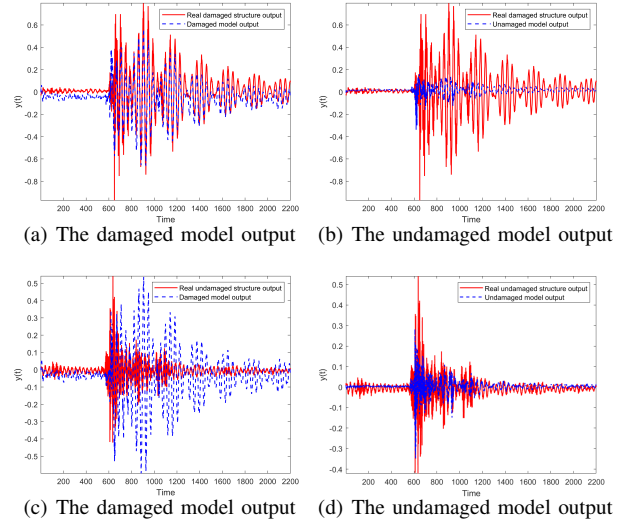

(a) The damaged model output
(b) The undamaged model output
(c) The damaged model output
(d) The undamaged model output

Fig. 6. Detecting Structural Damage in Building Sensor Data with Robust Echo State Networks: Output Results for the Cape Earthquake.

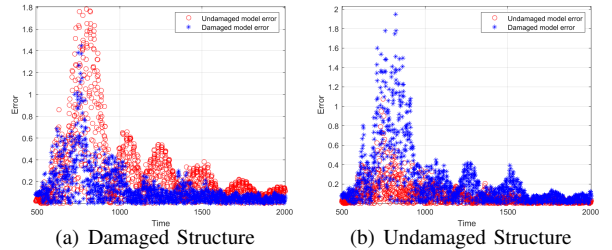
(a) Damaged Structure
(b) Undamaged Structure

Fig. 7. The errors corresponding to the damaged and undamaged structure models in the Northridge earthquake.

Hypertension and its Relationship with Socioeconomic Factors in Mexico Using Clustering Techniques

1st Obed Casillas-Baltazar
UPIITA-IPN
Intituto Politécnico Nacional
Mexico City, Mexico
ocasillas1800@alumno.ipn.mx

2nd Obdulia Pichardo-Lagunas
UPIITA-IPN
Intituto Politécnico Nacional
Mexico City, Mexico
opichardola@ipn.mx

3rd Bella Martinez-Seis
UPIITA-IPN
Intituto Politécnico Nacional
Mexico City, Mexico
bcmartinez@ipn.mx

Abstract—According to Ministry of Health in Mexico, Hypertension, commonly referred to as High Blood Pressure (HBP), continues to rank among the foremost ten causes of mortality in Mexico. This document describes the methodology for unearthing correlations between non-clinical variables and HBP, utilizing data clustering techniques in a data set derived from diverse Mexican institutions.

Index Terms—Hypertension,Socioeconomic Factors, Clustering Techniques

I. INTRODUCTION

Over the past twenty years, the global incidence of Non-Communicable Diseases (NCDs) has been on the rise. This issue is particularly pronounced in low- and middle-income countries, such as Mexico, where 75% of the deaths are associated with NCDs including Systemic Arterial Hypertension (SAH) [2]. Notably, systemic arterial hypertension has demonstrated a notable epidemiological shift in recent times. It is thought that together with the congenital factor of the disease, there are environmental and socioeconomic variables that can affect its development in the population.

II. PATTERNS OF ASSOCIATIONS WITHIN HYPERTENSION AND SOCIOECONOMIC DATA

A pattern means that the data are correlated, have a relationship, or are predictable. Data mining builds models to identify patterns among the variables in a data set. Some of these patterns are predictive (projecting future values), whereas others are explanatory (explaining the interrelationships among the variables) [1]. We integrated a data set from health institutions; then we study which socioeconomic indicators are more related to the registered hypertension cases in Mexico.

First, we will describe the data collection, selection, and integration of data related to hypertension and socioeconomic variables from health institutions in Mexico. Then, we propose a methodology using clustering in order to find

correlations between Hypertension and Socioeconomic Data. Finally, we evaluate them.

A. Data collection and selection for HBP and Prepossessing

The project encompasses two primary data categories: hypertension and socioeconomic variables that were collected from Mexican institutions. Patient records for hypertension are upheld by the Mexican government and health institutions such as IMSS, ISSSTE, and ENSANUT. Socioeconomic data originates from INEGI and the Mexican government's datos.gob open data platform.

Over 175 sources from ENSANUT, IMSS, ISSSTE, and the Health Sector were collected. We got 100,000+ records with 1,000+ attributes encompassing 2000 to 2020. Data sets include information on hypertension and socioeconomic indicators. A comprehensive exploration identified recurring entries, incomplete data, duplicated fields, and instances of invalid or unclear content. Data types were standardized. Data periodicity was evaluated, leading to the inclusion of databases with overlapping record periods into the final system. Finally, we got a unified data set in JSON format.

B. Clustering for correlation

Let $S = \{s_1, s_2, ..., s_m\}$ be the set of socioeconomic indicators; according to the selected data there are 46 different indicators, such that $m = 46$. The best number of clusters N was obtained by the elbow method. In this case $N = 5$, such that $C = \{C_1, C_2, C_3, C_4, C_5\}$. Each generated cluster C_i has a vector of weights $W_{c_i} = \{w_{c_i s_1}, w_{c_i s_2}, ..., w_{c_i s_m}\}$, where each $w_{c_i s_j}$ represents the weight (impact) of indicator s_j in the cluster c_i. Among them, the cluster c_x, such that $0 \leq x \leq N$, has the highest weight associated with the number of BPH-identified cases. On the other hand, the weights of each socioeconomic indicator s_j for each cluster c_i are evaluated. We obtain the cluster c_y, such that $0 \leq y \leq N$, in which the socioeconomic indicator s_j is dominant. If c_y is the same cluster as c_x, then it is established that the indicator s_j is strongly related to the cases of hypertension registered that year.

Two clustering algorithms were tested: k-means and hierarchical clustering. We used *k-means*, it groups objects into k groups depending on their characteristics. Hierarchical Clustering constructs a tree that represents the similar relationships between the different elements. We compare the indicators j of both algorithms and validate them with the Pearson correlation coefficient, which measures the statistical relationship, or association, between two continuous variables. It returns a value of between -1 and $+1$, where $+1$ represents the highest positive correlation.

III. EVALUATION OF THE CORRELATION OF BPH WITH SOCIOECONOMIC INDICATORS

We evaluate the correlation of BPH with socioeconomic indicators using clustering (k-means and HCA) and identifying the indicators with the highest weight for the cluster with more HAS; we also used Pearson Correlation Coefficient to identify the principal indicators in order to compare them with the results given by the clusters using Recall.

In order to compare the number of HBP cases with socioeconomic indicators, the required data sets were normalized. We test it for each of the 32 states in Mexico. We evaluated two clustering methods: Hierarchical Clustering (HCA) and k-means.

For k-means, the sum of squared distances from the center to the nearest cluster was calculated, which can be interpreted as the error range of the same model. For the HCA the distances between each observation in the hierarchical grouping, defined by the linkage matrix, were calculated; and the correlation of distances was obtained to determine the accuracy. The accuracy of HCA is measured through the generated distance matrix and the original data. The expected value for hierarchical clustering accuracy is 1, while for K-means error it is zero (it is important to remember that the latter is a sum of squared distances). The accuracy obtained was 0.7825 (78.25% success rate), i.e., for the 32 data vectors entered corresponding to each state, at least 25 were correctly clustered according to their socioeconomic characteristics. The classification results are presented in Table I.

TABLE I
CLUSTERING ALGORITHMS PRECISION EVALUATED BY THE THE ERROR RANGE FOR K-MEANS AND BY THE ACCURACY GIVEN BY CORRELATION OF DISTANCES FOR HCA

| | Precision | |
| | k-means | HCA |
Year Data Set	Error	Accuracy
2008	11769.69	0.7854
2010	10436.07	0.7902
2012	9526.41	0.7658
2014	9991.22	0.7684
2016	8348.98	0.7925
2018	9780.10	0.7932

Once the clusters are obtained, we evaluated the output vectors W for each c_i cluster and for each socioeconomic indicator s_j. In order to evaluate it, we compare the important indicators to the top five important indicators given by

PCC. The recall of this comparison is in Table II. For both algorithms, the recall is high because it is near 1.

TABLE II
RECALL OF SOCIOECONOMIC INDICATORS IN THE SAME CLUSTER (TWO CLUSTER ALGORITHMS) COMPARED WITH PCC

| | Recall | |
Year Data Set	k-means	HCA
2008	**0.9166**	0.8888
2010	**0.9000**	0.8750
2012	0.6000	**0.8888**
2014	0.9090	**1.0000**
2016	0.4444	**1.0000**
2018	**0.8888**	0.7500

Some of the most frequent socioeconomic indicators correlated with BPH are population vulnerable by income, non-poor and non-vulnerable population, contribution of educational backwardness to poverty, contribution of access to health care to poverty, contribution of access to social security to poverty, contribution of access to food to poverty, population affiliated to IMSS, population with private medical insurance, population with access to indirect social security health services and population food insecurity.

IV. CONCLUSIONS

The initial phase of this project involved establishing a unified database by amalgamating data from various publicly accessible sources concerning High Blood Pressure (HBP or hypertension) and socioeconomic indicators. Leveraging this standardized data enabled the prediction of registered hypertension cases and an assessment of the potential impact of non-medical factors on its development. Variables such as food insecurity, income vulnerability, and healthcare affiliation were examined for their potential influence.

REFERENCES

[1] Delen, Dursun. Predictive Analytics: Data Mining, Machine Learning and Data Science for Practitioners. FT Press, 2020.
[2] Secretaría de Salud, Enfermedades No Transmisibles Situación y Propuesta de Acción: Una Perspectiva desde la Experiencia de México, 2018. Primera edición. México: Secretaría de Salud, 2018.

An Ensemble Method for Applying Particle Swarm Optimization Algorithms to Systems Engineering Problems

Ken Hampshire
School of Engineering and Applied Science
George Washington University
Washington, DC
ken.hampshire@gwu.edu

Thomas Mazzuchi, D.Sc.
School of Engineering and Applied Science
George Washington University
Washington, DC
mazzu@gwu.edu

Shahram Sarkani, PhD
School of Engineering and Applied Science
George Washington University
Washington, DC
sarkani@gwu.edu

Abstract—As a subset of metaheuristics, nature-inspired optimization algorithms such as particle swarm optimization (PSO) have shown promise both in solving intractable problems, and in their extensibility to novel problem formulations due to their general approach requiring few assumptions. Unfortunately, a given algorithm requires detailed tuning of parameters and cannot be proven to be best suited to a particular problem class on account of the "no free lunch" (NFL) theorems. Using these algorithms in real-world problems requires exquisite knowledge of the many approaches and applying them based upon intuition. This research aims to present a unified view of PSO-based approaches from the perspective of relevant systems engineering problems, with the purpose to then elicit the best solution for any problem formulation in an ensemble learning approach. The central hypothesis of the research is that using the PSO algorithms found in literature to solve real-world optimization problems requires a general ensemble-based method for all problem formulations but a single implementation and solution for any instance. The main results will be a problem-based literature survey and a general method to find more globally optimal solutions for any systems engineering optimization problem.

Keywords—particle swarm optimization, ensemble learning, swarm intelligence

I. BACKGROUND

Particle swarm optimization (PSO) is one of the most promising global search algorithms used to solve optimization problems because the technique is simple to code and implement, has few parameters, and its flexibility allows for easy modification or hybridization with other metaheuristic approaches [1]. The technique was presented by Kennedy & Eberhart in 1995, and first initializes randomly generated particle (candidate solution) positions and velocities in the search space [2]. Then, these positions are evaluated relative to one another to determine the particle set best solution, and subsequent iterations move particles towards this best solution by updating each particle's position and velocity. For a given particle, the tradeoff between searching around itself and moving towards the set best is controlled by social and cognitive damping coefficients. This tradeoff is often referred to in literature as the balance between exploration and exploitation.

Despite its simplicity, PSO suffers from two drawbacks: premature convergence to local minima, and the inability to guarantee finding a global minimum. The latter problem is common to all metaheuristic approaches. Usually, searching stops after prescribed iteration or time limits. Much of the recent literature has been dedicated to improving the algorithm through three main approaches: identifying appropriate parameter settings, hybridizing PSO with other approaches, or using multiple particle sets (swarms) [1]. The last approach often turns to biology for inspiration in new ways to balance exploration and exploitation [3].

Novel PSO approaches in a paper usually highlight the inspiration, the algorithm approach, test the approach on benchmark optimization function sets such as CEC2005, compare performance to other PSO approaches, then illustrate the approach on one or more real-world problem formulations [4]. Yet Wolpert and Macready's "no free lunch theorems for optimization" (NFL) state that a given algorithm's performance on one class of problems is offset by its performance on another class [5]. This suggests that any two algorithms become computationally equivalent when averaged over all classes in a domain, creating a need for PSO approaches matched to problem classes.

The NFL theorems notwithstanding, a gap in the research literature is taking a problem-first approach to the study of PSO, specifically in the class of systems engineering problem formulations. Novel PSO papers often highlight their approach's relevance to real-world problems, which some review papers have aggregated when surveying and reconciling the various approaches [1], [6], [7]. Yet it remains difficult to apply the many PSO approaches to a particular systems engineering problem because doing so requires a unique understanding of those approaches and the ability to tune parameters for the given problem, while the NFL theorems suggest that relative performance is somewhat arbitrary. In short, most of the PSO literature is from the computational science perspective and assumes detailed optimization knowledge. While this is necessary, it is not sufficient to scale its adoption for use in systems engineering

on problems such as scheduling, allocation, network ontologies, or system design.

We hypothesize that systems engineering would benefit from systematically cataloging which PSO approaches have been used on what class of problems and providing a general method for their use when solving optimization problems in systems engineering. The general method is essential to account for the NFL theorems of optimization and defines a repeatable way to leverage PSO techniques to improve optimization in systems engineering.

II. Method

A. Literature Survey

Our literature survey will examine papers from journals which publish the preponderance of PSO research on systems engineering, including: *IEEE Transactions on Systems, Man, and Cybernetics; Expert Systems with Applications; IEEE Journal of Automatica Sinica; Complex & Intelligent Systems.* The goal of reviewing each paper is to understand and classify the PSO approach, then extract and categorize the sample problem formulations where that approach was used, especially as applied to systems engineering problems. There are two outputs of the literature survey: a catalog of PSO approaches and the types of problems they have been applied to as in Fig. 1, as well as a review paper which presents common systems engineering problem formulations and describes the PSO approaches applied to them from a problem-centric perspective.

B. Ensemble Optimization

The general method draws upon the PSO algorithm catalog produced by the literature survey to select from the most promising algorithms given knowledge of the problem at hand. Within a given class of problem, the method involves iteratively solving using each algorithm and returning the best solution from the set of candidate algorithms, similar to the ensemble learning approach found in machine learning.

Because PSO is a type of stochastic optimization, the best solution usually varies from run to run. Therefore, the method will be repeated for each problem multiple times (m runs) to ensure statistically significant results. The Friedman ranking test will be used to compare the relative performance of the various algorithms as in Table I.

Finding solutions through repeated iterations of multiple approaches is computationally expensive and not practical as the basis of a general method. Therefore, we hypothesize that there are two ways to construct the general method. First, if the relative performance of the algorithms is statistically insignificant, the bucket of algorithms will be used only as a diversity of solutions for a naïve problem formulation. Second, if the relative performance of the algorithms is statistically significant, the general ensemble method will be weighted to bias proportionally towards the best algorithms for a one-time solution to a naïve problem formulation.
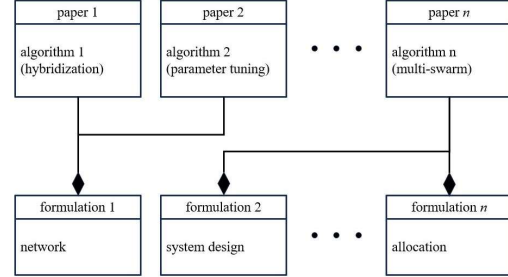

Fig. 1. Sample literature survey categorgization schema.

TABLE I. Data Organization Schema.

run		algorithm 1	algorithm 2	...	algorithm n
1	*mean*				
	std				
2	*mean*				
	std				
...					
m	*mean*				
	std				
rank					

References

[1] T. Shami et al., "Particle swarm optimization: a comprehensive survey," IEEE Access, vol. 10, pp. 10031-10061, 2022.

[2] J. Kennedy and R. Eberhart, "Particle swarm optimization," Proceedings of ICNN '95 – International Conference on Neural Networks, vol. 4, pp. pp. 1942-1948, November 1995.

[3] H. Minh, T. Sang-To, G. Theraulaz, M. Abdel Wahab, and T. Cuong-Le, "Termite life cycle optimizer," Exerpt Sys. Appl., vol. 213, p. 119211, March 2023.

[4] P. Suganthan et al., "Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization," KanGAL report, vol. 2005005, no. 2005, p. 2005, 2005.

[5] D. Wolpert and W. Macready, "No free lunch theorems for optimization," IEEE Trans. Evol. Comput., vol. 1, no. 1, pp. 67–82, April 1997.

[6] J. Nayak, H. Swapnarekha, B. Naik, G. Dhiman, and S. Vimal, "25 Years of Particle Swarm Optimization: Flourishing Voyage of Two Decades," Arch. Comput. Methods Eng., vol. 30, no. 3, pp. 1663–1725, April 2023.

[7] J. Tang, G. Liu, and Q. Pan, "A Review on Representative Swarm Intelligence Algorithms for Solving Optimization Problems: Applications and Trends," IEEE/CAA Journal of Automatica Sinica, vol. 8, no. 10, pp. 1627–1643, October 2021

Towards Interpretable Digital Twins for Self-Aware Industrial Machines

Adelson Santos S. Junior[1], João L. Vilar-Dias[2] and Fernando B. Lima-Neto[3]

Abstract— In this research, we introduce a methodology that combines digital twins and Particle Swarm Optimization (PSO) to improve real-time adaptability and interpretability in industrial systems. Using an industrial DC motor simulation as a case study, our approach involves creating a digital twin, performing online parameter estimation via PSO, and identifying unknown system components. The results, especially from scenarios like armature resistance degradation and unbalanced shaft conditions, highlight the digital twin's accuracy and adaptability. This work showcases the potential of our method for real-time monitoring and proactive maintenance in industrial applications.

I. BACKGROUND

The industrial domain has transitioned from manual operations to advanced automated systems since the third industrial revolution [1]. Effective control of dynamic systems necessitates understanding their dynamics [2]. However, real-world systems present challenges due to unpredictable factors, potentially leading to operational failures [3]. While adaptive models aim to enhance adaptability, they face complexities and require special instrumentation [4]. Digital Twins provide virtual representations of physical systems for various applications [5], [6]. Updated with real-time data, they offer more accurate predictions, enhancing machine self-awareness [6]. Particle Swarm Optimization (PSO) further augments this awareness by estimating model parameters [7]. The ability to interpret system models is crucial in the industrial domain for safety and efficiency [5] estimating non-instrumented parameters and adapting models using real-time data addresses system adaptability challenges [8]. PSO stands out for its efficiency in parameter estimation [7]. Hence, the aim of this work is to propose a methodology that enhances the interpretability and adaptability of Digital Twins using PSO.

II. PROPOSED METHODOLOGY

The proposed methodology seeks to establish an interpretable Digital Twin that integrates online parameter estimation and unknown components discovery using Particle Swarm Optimization (PSO). The methodology comprises three primary steps:

First, modeling the physical system: creating the digital twin based on mathematical equations that capture the system's physical components, ensuring accurate representation

and interpretability [10]. It emphasizes interpretability, parameterizability, and modularity.

Second, employing the Particle Swarm Optimization (PSO) approach for online parameter estimation: by leveraging acquired data to optimize the model's accuracy, focusing on minimizing the root mean square error (RMSE) between the model's predictions and the actual system outputs [9].

Third, utilizing PSO to identify unknown system components: addresses the discrepancy between the real system and the model by identifying and integrating unknown components through meta-models, typical mathematical models that impact system outputs. It aligns the model more closely with the real system by selecting an appropriate combination of these meta-models to complement the original model.

The final Self-Aware Digital Twin model combines the initial model with the added meta-models and parameters estimation, yielding a comprehensive representation that can adaptively mirror the real system's behavior.
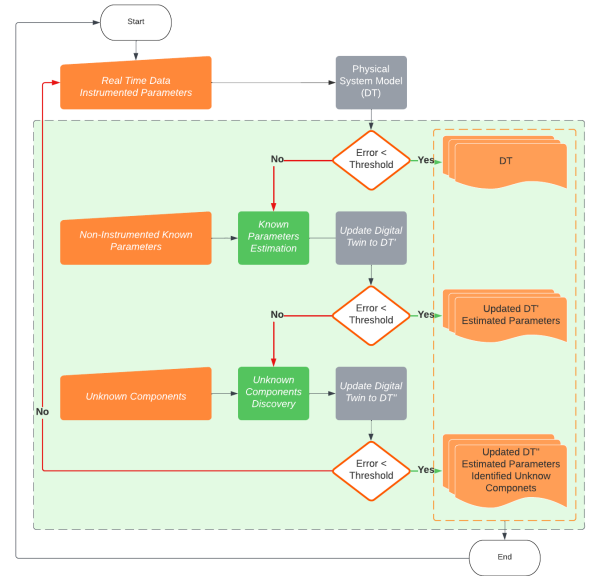

Fig. 1. Detailed flowchart of the proposed methodology

III. CURRENT RESULTS

The methodology has been examined through its application to an industrial DC motor simulation. This motor was defined by initial parameters: armature resistance $R_a = 6.5\Omega$, inductance $L_a = 0.673H$, moment of inertia $J = 0.001171\text{kg} \cdot \text{m}^2$, back electromotive force constant $K_e = 0.038\text{V} \cdot \text{s/rad}$, torque constant $K_t = 0.038\text{N} \cdot \text{m/A}$ and

[1]Adelson Santos S. Junior is with the Department of Computer Engineering, University of Pernambuco, Brazil assj2@ecomp.poli.br
[2]João L. Vilar-Dias is with the Department of Computer Engineering, University of Pernambuco, Brazil jlvd@ecomp.poli.br
[3]Fernando B. Lima-Neto is with the Department of Computer Engineering, University of Pernambuco, Brazil fbln@ecomp.poli.br

damping coefficient $D = 0.00143 \text{N} \cdot \text{m/s} \cdot \text{rad}$, reflecting its wide usage in various industrial processes [9].

For optimization, the PSO technique from the 'pymoo' library [11] was employed with defining parameters: a swarm size of 15, a maximum iteration limit of 1000, an inertia weight of 0.9, and both cognitive and social parameters set at 2.0.

In the Armature Resistance Degradation scenario, R_a is decreased along time from 6.5Ω to about 5.6Ω using an exponential decay $-0.044 * \exp(0.003 * t) + 0.044$ each time step initiating at $160s$. In this case, there was a pronounced alignment between the Digital Twins' predictions and the simulated data. This congruence is clearly visualized in Figure 2 which shows the model's estimation percentage error.

To simulate an Unbalanced Shaft, deviations are introduced to the rotor speed. A sinusoidal error is added to the speed calculation at each time step. The amplitude of the error is set as $5rad/s^2$ which is multiplied by the simulation sample time. The frequency is chosen to be fixed at $10rad/s$. In this case, the parameter estimation is not enough to guarantee the model prediction performance as shown in Figure 3. Once the unknown components discovery takes place, the updated model is capable of better predicting and representing the plant behavior. Figure 4 depicts the model prediction and the prediction error of the new updated model for a $2s$ estimation time window. It shows that the maximum absolute error is reduced from $0.5rad/s$ to less than $0.02rad/s$.

The results underscore the robustness of the proposed methodology, especially Steps 2 (Parameter Estimation and Digital Twin Update) and 3 (Unknown Components Discovery). By leveraging these steps, the digital twin not only mirrored the real system's behavior but also demonstrated the ability to adapt to deviations and anomalies. The methodology's emphasis is on real-time parameter estimation and adaptability. It showcases a model that is not only accurate but also self-aware, capable of identifying and rectifying discrepancies in real time. These positions are crucial for industries, ensuring system reliability.
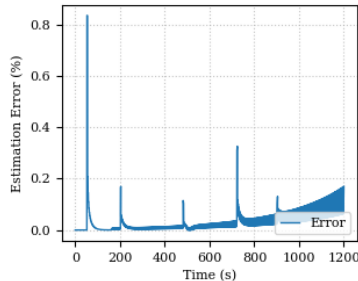

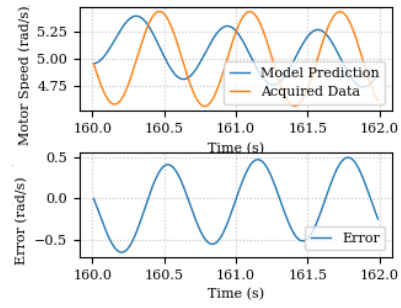Fig. 2.   Estimated percentage error for the Armature Resistance.

Fig. 3.   Predicted and acquired data for output speed before Unknown Components Discovery
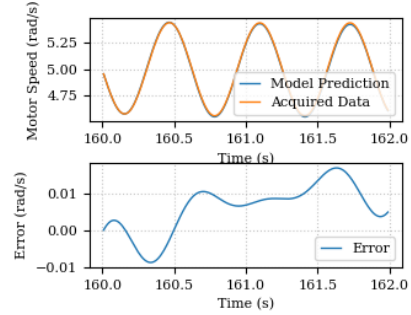

Fig. 4.   Predicted and acquired data for output speed after Unknown Components Discovery

REFERENCES

[1] M. Bahrin, F. Othman, N. Azli, and M. Talib, "Industry 4.0: A review on industrial automation and robotic," Jurnal Teknologi, vol. 78, no. 6-13, pp. 137-143, 2016. doi: 10.11113/jt.v78.9285

[2] R. J. Brown, *A Modern Introduction to Dynamical Systems*, Oxford University Press, 2018.

[3] A. Kuhnle, JP. Kaiser, F. Theiß, N. Striker and G. Lanza, "Designing an adaptive production control system using reinforcement learning," *Journal of Intelligent Manufacturing*, vol. 32, no. 3, pp. 855-876, Jul. 2021, doi: 10.1007/s10845-020-01612-y.

[4] L. Whitcomb, S. Arimoto, T. Naniwa, and F. Ozaki, "Adaptive model-based hybrid control of geometrically constrained robot arms," *IEEE Trans. Robotics Autom.*, vol. 13, no. 1, pp. 105-116, Feb. 1997.

[5] A. Fuller, Z. Fan, C. Day and C. Barlow, "Digital Twin: Enabling Technologies, Challenges and Open Research," in *IEEE Access*, vol. 8, pp. 108952-108971, 2020, doi: 10.1109/ACCESS.2020.2998358.

[6] B.R. Barricelli, E. Casiraghi, and D. Fogli, "A Survey on Digital Twin: Definitions, Characteristics, Applications, and Design Implications," *IEEE Access*, vol. 7, pp. 167653-167671, 2019.

[7] J. Kennedy and R. Eberhart, "Particle swarm optimization," *Proceedings of ICNN'95 - International Conference on Neural Networks*, Perth, WA, Australia, 1995, pp. 1942-1948 vol.4, doi: 10.1109/ICNN.1995.488968.

[8] B. Ljung, *System identification: theory for the user*, Prentice Hall, Upper Saddle River, NJ, USA, 1999.

[9] A. Polsena, Y. Kongjeen and S. Watcharakhup, "Identifying Parameter and PI Tuning of DC Motor for Predict Behavior based on PSO," *2021 9th International Electrical Engineering Congress (iEECON)*, Pattaya, Thailand, 2021, pp. 97-100, doi: 10.1109/iEECON51072.2021.9440341.

[10] O. G. Brylina, N. N. Kuzmina and K. V. Osintsev, "Modeling as the Foundation of Digital Twins," 2020 Global Smart Industry Conference (GloSIC), Chelyabinsk, Russia, 2020, pp. 276-280, doi: 10.1109/GloSIC50886.2020.9267812.

[11] Blank, J. & Deb, K. pymoo: Multi-Objective Optimization in Python. *IEEE Access*. **8** pp. 89497-89509 (2020)

Real Time Continuous Image Stitching Algorithm Based on SIFT

YANG Rui-jun
School of Computer Science &
Information Engineering
Shanghai Institute of Technology
Shanghai China
yangruijun@sit.edu.cn

ZHANG Chu
School of Computer Science &
Information Engineering
Shanghai Institute of Technology
Shanghai China
sunnyzhang2020@163.com

CHENG Yan
School of Criminal Justice
East China University of Political
Science and Law
Shanghai China
chengyan@ecupl.edu.cn

Abstract-**Image stitching technology has application scenarios in many fields. This algorithm achieves the acquisition of simultaneously synthesized images, using the RGB module of the Intel Realsense D435 camera for image acquisition. Firstly, a raw image is created to store the final result, and one image is collected at 100ms intervals each time. Images with similarity less than 5/8 are taken and saved. Use the SIFT scale invariant feature detection algorithm to extract image feature points from the collected images, use RANSAC to extract feature points, use the random sample consistency algorithm to filter effective points, and calculate the homography transformation matrix. Synthesize every two collected images into one image and overlay it at the corresponding position in the resulting image. Through the experiment in this article, the average time for single acquisition and synthesis is 70ms, achieving the real-time goal. The similarity between the experimental group and the control group can reach 70%, and the resolution has been increased by 1.65 times, achieving the goal of continuous splicing.**

Keywords: image stitching; SIFT; RANSAC; real-time; continuous

I. INTRODUCTION

In recent years, image stitching technology has become a hot topic. Image stitching technology is the process of combining two or more images, based on one image, comparing other images with this image, and covering the same area to generate a new image. The process of image stitching is mainly divided into three steps: image matching [1], image transformation [2] and image synthesis [3].

The current commonly used technical methods usually use the method of collecting first and then processing. The synthesis results are not displayed together with the collection, so they are lacking in real-time and scalability.

This paper proposes a method that uses the SIFT algorithm for feature point extraction, uses the RANSAC algorithm to calculate the homography matrix transformation method for image transformation, and performs image synthesis after processing. On this basis, by creating a result map, the synthesized results are overlaid on the result map one by one. The result map is matched according to the size of the result of image synthesis. If it exceeds the range, it will be extended. In this way, continuous image synthesis is realized.

II. Correlation Algorithms

A. Feature Point Detection Algorithm

This paper uses the SIFT algorithm, which was published by Professor David Lowe in 1999 [4] and further improved in 2004 [5].

Using the SIFT algorithm, the scale space should first be constructed by using the Gaussian kernel function to blur the image, that is, to combine several local pixels into one pixel. The Gaussian function is:

$$G(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2}{2\sigma^2}} \tag{1}$$

where σ is the standard deviation of the normal distribution, the calculation formula for the $m \times n$ convolution kernel is:

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x-0.5m)^2+(y-0.5n)^2}{2\sigma^2}} \tag{2}$$

LOG is Laplacian of Gaussian, DOG is Difference of Gaussian, where DOG is used to approximate LOG, and DOG can provide more stable image features than LOG.

The SIFT algorithm solves the scale invariance through the scale space, and the expression of the scale space $L(x,y,\sigma)$ is：

$$L(x,y,\sigma) = G(x,y,\sigma) * I(x,y) \tag{3}$$

In the formula, $G(x,y,\sigma)$ is a Gaussian function; "*" is a convolution operation, and $I(x,y)$ is the original image.

The Gaussian pyramid is constructed by calculating the volume of the Gaussian kernel function $G(x,y,\sigma)$ and $I(x,y)$ under the constantly changing σ Product calculated.

$$D(x,y,\sigma) = \big(G(x,y,k\sigma) - G(x,y,\sigma)\big) * I(x,y)$$
$$= L(x,y,k\sigma) - L(x,y,\sigma) \tag{4}$$

The DOG pyramid is obtained by subtracting two adjacent images of the same group of images in the tower.

On the DOG pyramid, it is necessary to compare the point of 29 pixels need to be compared points, so as to ensure that extreme points can be detected in both image space and scale space. Only if a pixel in this layer is larger than all these adjacent pixels or smaller than all adjacent pixels, it will be selected as an extremum point.

B Image Stitching Method

This article uses a random sampling consensus algorithm proposed by Fischler and Bolls in 1981. The RANSAC algorithm randomly selects 4 matching pairs from the matching data

set (these 4 matching pairs are required to be non-collinear), and obtains a homography matrix by calculating the matching data.

III. Algorithm implementation process

Collect two images successively, use the SIFT algorithm for key point matching, use the matched key points to calculate the transformation matrix through the RANSAC algorithm, perform image transformation through this transformation matrix, and synthesize the two images into one image.

A. Acquire two images in real time

In order to realize the real-time synthesis function, this paper closes the previous data stream every 100 milliseconds, stores the image *img1* obtained before closing, and then re-reads the camera data stream to obtain the image *img2*,

In practical applications, the camera may move slightly and should not continue to acquire new images at this time. This article uses the method of comparing image hash values to preserve images with a similarity of less than 5/8.

B. Synthesize two images into one image

Use the SIFT algorithm to obtain the key points *KP1* and *KP2* of two images. Using the *knnMatch* method, find the point m closest to the *img1* key point and the second closest point n in the *img2* key point, and match them together to form a match. Use the ratio of Euclidean distance to match the key points in m and n. When it is less than 0.5, the appropriate key points are retained, otherwise they will be discarded:

$$m.distance < minValue \times n.distance \quad (5)$$

Next, use the RANSAC method to calculate the transformation matrix *M*. Transform *img2* through *M* to obtain the synthesized *resultImg*.

C. Into the result image

When the camera moves to the left, it corresponds to the negative direction; when the camera moves to the right, corresponds to the positive direction. This movement value is superimposed in the positive direction. If it does not exceed the boundary, it is considered to be only moving to the right. If it exceeds the boundary, it is considered that the right side needs to increase the range.

IV. Experiment and Result Analysis

A. Exp data collection and processing

In this experiment, all images of the target are obtained by moving the camera horizontally. The composite target is a tiled image of 600mm*245mm.

This experiment uses a solid color background in which to place the target image. In the process of synthesizing the result image by means of translation, the key points collected by the SIFT algorithm are all on the target picture, and the captured or synthesized image is rectangularly cropped, and only the part of the target picture is kept for comparison.

Randomly select two of the images to match and synthesize the results as shown in Figure 1:
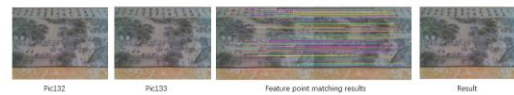

Fig.1 Experimental process

Directly shooting and collecting complete images as the result, therefore the results of this experiment are used as the control group. The experimental group completed the overall shooting through 55 images and performed the synthesis operation, as shown in Figure 2.

Fig.2 Control group(up) and experiment(down)

C. Analysis of results

The average duration of a single synthesis is 70ms, and the total duration of multiple synthesis varies linearly to achieve real-time synthesis goals.

For the post-experimental results, use 16*16 hash operations to calculate the similarity with the first set of results, that is, the original image, and compare the resolution change ratio.

The algorithm proposed in this paper has a similarity of 70% between the synthetic result and the original image. The resolution is also increased from the original 629*263 to the highest 1113*448, the length is increased to 1.76 times, and the width is increased to 1.70 times.

REFERENCES

[1] M. Brown, D.G. Lowe, Automatic panoramic image stitching using invariant features[J]. Int. J. Comput Vis. 2007, 74 (1):59–73.

[2] S.Peleg, J.Herman, Panoramic mosaics by manifold projection, Computer Vision and Pattern Recognition[C]// 1997. Proceedings., 1997 IEEE Computer Society Conference on. 1997:338–343.

[3] M.-S. Su, W.-L. Hwang, K.-Y. Cheng, Analysis on multiresolution mosaic images[C]// IEEE Trans. Image Process. 2004, 13 (7): 952–959. 2022,48(8):159-162

[4] Lowe D G. Distinctive Image Features from Scale-Invariant Keypoints[J]. International Journal of Computer Vision, 2004, 60(2):91-110.

[5] M.A. Fischler and R.C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography[J]. Communications of the ACM, 1981,24(6):381–395

Synthetic Generation of Pneumonia Images using CycleGAN Model

Gerardo Lugo-Torres
Center for Computing Research (CIC)
Instituto Politécnico Nacional
Mexico City, Mexico
glugot2022@cic.ipn.mx

Diego A. Peralta-Rodríguez
Computational Cognitive Sciences Lab. - CIC
Instituto Politécnico Nacional
Mexico City, Mexico
dperaltar1100@alumno.ipn.mx

José E. Valdez-Rodríguez
Computational Cognitive Sciences Lab. - CIC
Instituto Politécnico Nacional
Mexico City, Mexico
jvaldezr2018@cic.ipn.mx

Hiram Calvo
Computational Cognitive Sciences Lab. - CIC
Instituto Politécnico Nacional
Mexico City, Mexico
hcalvo@cic.ipn.mx

Abstract—The utilization of generative models in image synthesis has become increasingly prevalent. Synthetic medical imaging data is of paramount importance, primarily because authentic medical imaging data is scarce, costly, and encumbered by legal considerations pertaining to patient confidentiality. Consent forms are typically required from patients in order to utilize their data for publication in medical journals or educational purposes. Consequently, the accessibility of medical data for general public research is limited. Synthetic medical images offer a potential resolution to these issues. The predominant approaches primarily assess the quality of images and the degree of resemblance between these images and the original ones employed for their generation. In this study, we employ a CycleGAN model to produce artificial images depicting several types of pneumonia, including general, bacterial, and viral pneumonia. We then evaluate the performance of these synthetic images by comparing them with ratings made by three respiratory care professionals. Consequently, a range of pneumonia pictures were acquired, exhibiting diverse levels of performance, ranging from being easily identified as false to being correctly identified as real in over 80% of cases.

Index Terms—Synthetic, Chest x-ray, Cyclic, Generative Adversarial Network, Pneumonia,Image-to-image,Translation

Recent research has emphasized generative models for image synthesis [1]. AI's growing role in medical research, especially in processing images, text, and sound, demands significant data. Collecting this data involves time, resources, and obtaining enough annotated data for effective CNN training.

This study explores using a CycleGAN [2] to produce chest X-ray images of various pneumonia types with quality comparable to "real" data for AI medical models. Although previous research has utilized generative models for medical image creation [1], few [3] delve into interpreting metrics determining when generated images can be deemed natural for AI medical use.

Our aim is to examine the correlation between the Frechet Inception Score (FID) and Inception Score (IS) [4] with respiratory care expert evaluations, gauging the "realness" of the generated images. This work contributes by: 1) Comparing FID and IS metrics with pneumonia images from a CycleGAN model over 100 training epochs; and 2) presenting evidence that these metrics don't align with expert evaluations on image authenticity.

The CycleGAN architecture is different from other GANs because it contains 2 mapping function (G and F) that acts as generators and their Discriminators (D_x and D_y): The generator mapping functions are: $G : X \to Y$ $F : Y \to X$ where X is the input image distribution and Y is the desired output distribution. The cost function is the sum of adversarial loss and cyclic consistent loss:

$$L(G, F, D_x, D_y) = L_{\text{advers}}(G, D_y, X, Y) + L_{\text{advers}}(F, D_x, Y, X)$$
$$+ \lambda L_{\text{cycl}}(G, F, X, Y)$$

with an objective function with the form of:

$$\min_{G,F} \max_{D_x, D_y} L(G, F, D_x, D_y)$$

We trained a CycleGAN, whose architecture is described in -B, for 100 epochs with 3 datasets: 1341 normal X-ray images, 2531 bacterial pneumonia images and 1345 viral pneumonia images. A total of 1,341 synthetic images of general pneumonia (GP), bacterial pneumonia (BP), and viral pneumonia (VP) were generated at a different number of training:25, 50, 75, and 100 epochs. For each set of images, its corresponding FID and IS values were calculated.

Three questionnaires, each containing 100 images, were sent to three respiratory care experts. The images were divided into four sections for evaluation. Out of the 100 images, 80 were generated, and 20 were real. In the questionnaires, they were given the option to choose if the image they saw was real or fake and if the image corresponded to general pneumonia, bacterial pneumonia, viral pneumonia, or did not correspond to pneumonia.

Table I
QUANTITATIVE RESULTS.

Dataset	Epochs	FID	IS
General Pneumonia	25	64.629	1.8011
	50	93.1049	1.8217
	75	106.5786	1.8568
	100	86.5787	2.3461
Bacterial Pneumonia	25	68.9245	2.1217
	50	64.5383	1.9673
	75	78.4502	1.824
	100	67.4041	1.296
Viral Pneumonia	25	54.4719	2.2069
	50	63.9624	2.2485
	75	77.9493	2.489
	100	60.9601	2.5063

Table II
PERCENTAGE OF SAMPLES IDENTIFICATION.

Dataset	Epochs	Expert 1		Expert 2		Expert 3	
		SR*	RR*	SR	RR	SR	RR
GP	25	0.55	0.8	0.6	0.8	0.6	0.6
	50	0.35	0.8	0	0.8	0.35	0.8
	75	0.45	1	0.4	0.8	0.35	0.8
	100	0.45	0.6	0.3	0.6	0.45	0.8
BP	25	0.7	0.6	0.6	1	0.7	0.6
	50	0.7	0.8	0.45	0.6	0.65	0.4
	75	0.8	0.2	0.95	1	0.95	0.8
	100	0	0.6	0.2	2	0.15	0.8
VP	25	0.55	1	0.6	1	0.35	1
	50	0.35	1	0.8	0.6	0.35	0.6
	75	0.68	0.8	0.55	1	0.25	1
	100	0.55	0.6	0.7	1	0.35	1

*SR:synthetic image and class identified as real , RR: real image and class identified as real.

The results obtained give us evidence that the images generated score either FID or IS do not translate into their ability to pass as images of real pneumonia.

A. Conclusions

In this work, we performed a comparison between scores and the subjective assessment of experts. We didn´t see a relationship between the best FID, IS, and the best assess expert score images. This will suggest the need for a proposal of a better score that will allow the use of synthetic images more reliably and helpfully. A DCNN model has been proposed for evaluating the quality of the images generated, an Augmenteed Cycle GAN to improve image quality, and sought radiology experts to include more and better assessments.

B. Dataset & Code

The code used in this paper and the dataset is available at: https://github.com/Lugo1025/PneumoCGAN.

REFERENCES

[1] D. I. Morís, J. de Moura, J. Novo, and M. Ortega, "Cycle generative adversarial network approaches to produce novel portable chest x-rays images for covid-19 diagnosis," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 1060–1064.

[2] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2223–2232.

[3] B. Segal, D. M. Rubin, G. Rubin, and A. Pantanowitz, "Evaluating the clinical realism of synthetic chest x-rays generated using progressively growing gans," *SN Computer Science*, vol. 2, no. 4, p. 321, 2021.

[4] A. Borji, "Pros and cons of gan evaluation measures: New developments," *Computer Vision and Image Understanding*, vol. 215, p. 103329, 2022.
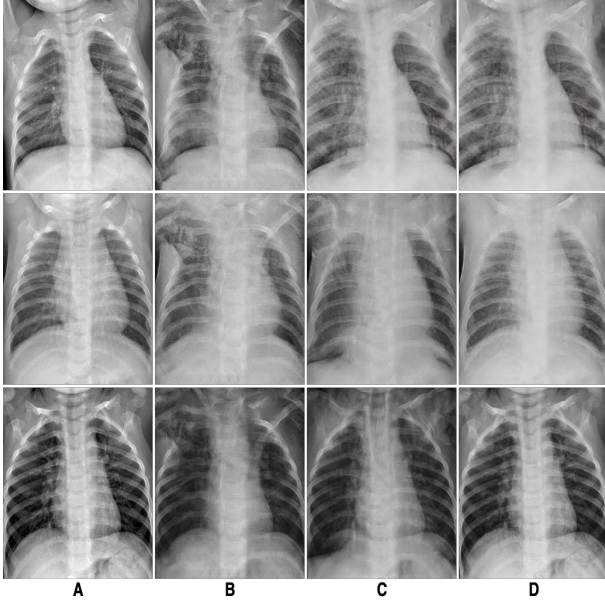
Figure 1. Results obtained from our CyclicGAN model, A)Original Image, B)General pneumonia, C)Bacterial pneumonia, D)Viral pneumnonia

For the GP generated images the best expert assessment average of 0.58 was with 25 epochs of training and match also with the best FID but not IS score of its group. The best BP generated images was with 75 epochs of training and got an average expert assessment of 0.9 which which does not match the best FID or IS score of its group, overall this group produce the best score assessment images by the experts of all. And for the VP generated images the best expert assessment average of 0.53 was with 100 epochs of training which does not match the best FID or IS of its group.

Time Series Prediction Based on Randomly Weighted Neural Networks

Xizhao Wang
*College of Computer Science
and Software Engineering,
Shenzhen University,
Shenzhen, China*
xzwang@szu.edu.cn

Qin Wang
*College of Computer Science
and Software Engineering,
Shenzhen University,
Shenzhen, China*
wangqin2021@email.szu.edu.cn

Qiang Liu
*College of Computer Science
and Software Engineering,
Shenzhen University,
Shenzhen, China*
liuqiang2022@email.szu.edu.cn

Abstract—One of the most frequently used models for time series prediction is the Long Short Term Memory (LSTM). LSTM can leverage the past patterns to efficiently forecast the future observations but it is often criticized as very computationally expensive due to the iterative training. In this paper, to reduce the computational workload and improve the prediction performance of time series, we propose a novel auto-regression framework based on Random Vector Functional Link (RVFL). The new framework offers a lighter network structure with higher training efficiency compared to LSTM-based approaches. It is a new attempt to utilize randomized learning algorithms for time series prediction, providing valuable insights for developing faster and more efficient models in the future.

Index Terms—time series prediction, auto-regression, long short-term memory, randomly weighted neural networks

I. Introduction

Time series data consists of a sequence of observations ordered in time, and time series prediction aims to forecast future observations based on past patterns [1]. Auto-regression (AR) is a commonly used approach for time series prediction. However, due to the limited approximation ability of polynomials, AR struggles to handle complex nonlinear time series. With the advent of deep learning, LSTM-based methods have emerged as promising approaches for time series prediction [2], [3]. Nonetheless, the training of LSTM models can be computationally inefficient due to the large number of parameters in the gate cells. Additionally, as the number of network layers increases, LSTM may pose challenges in terms of hardware requirements and energy consumption.

To overcome these challenges, we propose a novel moving auto-regression framework called MAR-RVFL, based on Random Vector Functional Link (RVFL). RVFL is a type of randomly weighted neural network, where the weights between the input layer and hidden layer are randomly generated, and the weights between the hidden layer and output layer are calculated using matrix inverse operations [4]. RVFL's simple structure and non-iterative training mechanism enable reliable generalization and fast training speeds [5]. By combining RVFL with a specially designed sliding window mechanism, MAR-RVFL effectively learns the underlying patterns in time series data. Compared to LSTM, MAR-RVFL demonstrates higher training efficiency and robustness, providing valuable insights for more powerful time series prediction models.

II. Preliminarie

A. Auto-regression (AR)

AR is a method used for predicting future observations in a time series by analyzing the relationships between past observations. It assumes that the $(k + 1)$th observation can be expressed as a linear combination of the previous k observations, as shown in Equation (1).

$$\hat{x}_{k+1} = c + \varphi_1 x_1 + \varphi_2 x_2 + \ldots \varphi_k x_k + \varepsilon_{k+1} \qquad (1)$$

where $\hat{x}_{k+1}$ is the evaluated observation at the $k+1$ th moment, φ_i is the influence factor of the i th moment, x_i is the i th observation, and c is a noise term. The objective is to estimate φ_i using methods such as least squares or gradient descent.

B. Random Vector Functional Link (RVFL)

RVFL is typical single hidden layer neural network. In RVFL, the weights between input layer and hidden layer are randomly assigned, the weights between input layer and output layer, as well as weights between hidden layer and the output layer (represented by β) are calculated by solving the optimization problem in Eq.(2).

$$Minimize : \|H\beta - Y\| \qquad (2)$$

The solution to the above problem is defined as Eq.(3)

$$\beta = H^{\dagger} Y \qquad (3)$$

where $H^{\dagger}$ is the Moore-Penrose generalized inverse of H [6].

III. Proposed Method

A. Moving Auto-regression RVFL(MAR-RVFL)

Assuming here we intend to predict future k moments' observations according to the past k observations in a time series. The original training data is a one-dimension sequence with a length of T.

$$S = \{s_1, s_2, s_3 \ldots s_t \ldots s_T\} \qquad (4)$$

where s_t is the observation at moment t. According to the defination of the task, the sequence can be transformed into features matrix $X \in R^{z \times k}$ and labels matrix $Y \in R^{z \times k}$ ($z = (T - k)/k$ is the amount of instances). Then we have:

$$x_i = [s_{(i-1)k+1}, s_{(i-1)k+2}, \ldots s_{ik}], y_i = [s_{ik+1}, s_{ik+2}, \ldots s_{ik+k}] \qquad (5)$$

Based on the approach presented in [7], we first initialize the weights and biases between the input layer and the hidden

layer (W and b) with random values. The weights between the input layer and the output layer are ignored.Then we construct a initial window with a size of u:

$$U_X = [x_1, x_2, x_3 \ldots x_u], U_Y = [y_1, y_2, y_3 \ldots y_u] \quad (6)$$

The output of hidden layer is $H = g(U_X \cdot W + b)$, where g is the activation function. According [7], the weights betweent hidden layer and output layer β can be calculated as Eq.(7).

$$\beta = H^{\mathrm{T}} \left(\frac{I}{C} + HH^{\mathrm{T}} \right)^{-1} U_Y \quad (7)$$

We believe the latter observations in the time series show more relations with the observations in the future. Consequently, we use the latter observations in the window to further optimize the random weights W and b through Back Propagation (BP) algorithm.

$$W = W + \alpha \cdot \frac{\delta J(W, \beta, U \cdot M)}{\delta W} \quad (8)$$

where M is a mask vector in which the former m elements are valued 0 and the latter $u - m$ elements are valued 1, J denotes the training loss, and α is the learning rate.

The process of MAR-RVFL involves sliding the window U_X and U_Y over the input and output instances with a step size of L. After each slide, the weights β are updated using Equation (10), which aims to minimize the training error under the new window while preserving the patterns learned from previous iterations. Assuming the slided new window is:

$$U_X^* = [x_{1+L}, x_{2+L}, \ldots x_{u+L}], U_Y^* = [y_{1+L}, y_{2+L}, \ldots y_{u+L}] \quad (9)$$

To adjust β for the new window and retain patterns in former iterations, we optimize β based on the following new object.

$$Minimize : \left\| H \cdot \beta^* - U_Y^* \right\| \, and \, \left\| \beta^* - \beta \right\| \quad (10)$$

where β^* denotes the optimized β. The first term represents the training error under new window, which is minimized in MAR-RVFL for fitting new window's observations. The second term $\|\beta^* - \beta\|$ is the L2-norm for updating β^*, this term is minimized for inheriting former patterns. Based on lagrange method of multipliers, β^* is calculated as Eq.(11).

$$\beta^* = H^{\mathrm{T}} \cdot \left(\frac{I}{C} + H \cdot H^{\mathrm{T}} \right)^{-1} \cdot (U_Y^* - H \cdot \beta) + \beta \quad (11)$$

Then we again intend to optimize random weights based on latter observations in the new window. The MAR-RVFL method continues to slide the window and update the weights until it has iterated over the desired number of times. By doing so, it gradually incorporates information from different parts of the time series, capturing temporal dependencies and improving the prediction accuracy.

IV. Experiment

The proposed MAR-RVFL method is compared with LSTM on three real-word databases (DB1: daily temperature, DB2: sinewave, DB3: milk production). To ensure fairness, we use one LSTM layer as hidden layer in LSTM, and set the number of hidden nodes in both LSTM and MAR-RVFL as 2,000. Fig.1 shows the comparison for all three databases. It can be observed that MAR-RVFL outperforms LSTM in DB1
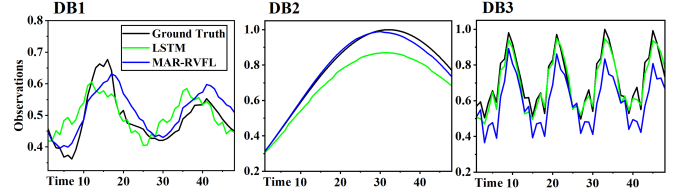

Fig. 1. Predicted and actual observations.

and DB2, showing better prediction ability. In DB3, LSTM performs slightly better than MAR-RVFL.

Fig.2 shows the MSE comparison between LSTM and MAR-RVFL for the three databases. It can be observed that MAR-RVFL achieves lower MSE values than LSTM in DB1 and DB2, indicating superior prediction performance. Assuming we have z instances after the pretreatment
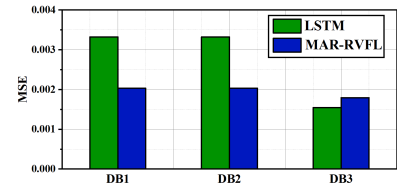

Fig. 2. Testing MSE of LSTM and MAR-RVFL on different databases.

Table.I shows the time cost during training. It can be seen that MAR-RVFL has a significantly shorter time compared to LSTM, making it computationally more efficient.

TABLE I
Time cost during training in LSTM and MAR-RVFL

Model	LSTM	MAR-RVFL
Time Cost (s)	192.7	41.9

V. Conclusion

In conclusion, the MAR-RVFL proposed in this paper offers a promising approach for time series prediction. By integrating RVFL into auto-regression, MAR-RVFL demonstrates competitive prediction ability compared to the widely used LSTM. Additionally, it exhibits faster training times, making it an efficient option for time series analysis.

References

[1] N. T. Koh, A. Sharma, J. Xiao, X. Peng, and W. L. Woo, "Solar irradiance forecast using long short-term memory: A comparative analysis of different activation functions," in *2022 IEEE Symposium Series on Computational Intelligence (SSCI)*. IEEE, 2022, pp. 1096–1101.

[2] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[3] G. Van Houdt, C. Mosquera, and G. Nápoles, "A review on the long short-term memory model," *Artificial Intelligence Review*, vol. 53, pp. 5929–5955, 2020.

[4] Y.-H. Pao and Y. Takefuji, "Functional-link net computing: theory, system architecture, and functionalities," *Computer*, vol. 25, no. 5, pp. 76–79, 1992.

[5] Q. Wang, J. Liu, W. Guo, and X. Wang, "Evolving stochastic configure network: A more compact model with interpretability," *Information Sciences*, vol. 639, p. 119006, 2023.

[6] D. Serre, "Matrices: Theory & applications additional exercises," *L'Ecole Normale Supérieure de Lyon*, 2001.

[7] D. Wang and M. Li, "Stochastic configuration networks: Fundamentals and algorithms," *IEEE transactions on cybernetics*, vol. 47, no. 10, pp. 3466–3479, 2017.

Seed Kernel Counting using Domain Randomization and Object Tracking Neural Networks

Venkat Margapuri
Department of Computer Science
Villanova University
Villanova, PA
vmargapu@villanova.edu

Prapti Thapaliya
Department of Computer Science
Villanova University
Villanova, PA
pthapali@villanova.edu

Mitchell Neilsen
Department of Computer Science
Kansas State University
Manhattan, KS
neilsen@ksu.edu

Abstract—**High-throughput phenotyping (HTP) of seeds is the comprehensive assessment of complex seed traits and the measurement of parameters that form more complex traits [1]. The key aspect of seed phenotyping is cereal yield estimation. While mechanized seed kernel counters are available in the market currently, they are often priced high and sometimes outside the range of small scale seed production firms' affordability. The development of object tracking neural network models such as You Only Look Once (YOLO) enables computer scientists to design algorithms that can estimate cereal yield inexpensively. The key bottleneck with neural network models is that they require a plethora of labelled training data before they can be put to task. We demonstrate that the use of synthetic imagery serves as a feasible substitute to train neural networks for object tracking. Furthermore, we propose a seed kernel counter that uses a low-cost mechanical hopper, trained YOLOv8 neural network model, and object tracking algorithms on StrongSORT and ByteTrack to estimate cereal yield from videos. The experiment yields a seed kernel count with an accuracy of 95.2% and 93.2% for Soy and Wheat respectively using the StrongSORT algorithm, and an accuray of 96.8% and 92.4% for Soy and Wheat respectively using the ByteTrack algorithm.**

Index Terms—**YOLOv8, Artificial Intelligence, Domain Randomization, Object Tracking, Seed Counter**

I. Introduction

The advent of technology in agriculture commenced over a century ago, and several studies have been conducted since the 1990s to improve production efficiency [2]. High-throughput Phenotyping (HTP) of seeds is the comprehensive assessment of complex seed traits and the measurement of parameters that form more complex traits [1]. Currently, seed production firms have to use expensive mechanized seed counting machinery to pack seed kernels by count. This paper demonstrates leveraging videos of seed kernels rolling down a platform to estimate seed kernel count using low-cost hardware components (described in section II) and the object tracking neural network model, You Only Look Once (YOLO). Supervised neural network models require a plethora of labelled information to train for tasks. However, labeled training data is not always readily available for entities such as seed kernels. We demonstrate that the use of synthetic image datasets, generated following the principles of Domain Randomization [3]–[5], is a feasible alternative to train neural network models.

II. Related Work

Neilsen et al. [6] proposed an image processing algorithm to conduct seed kernel counting from videos. It is based on tracking each of the seed kernels as they flow down a backlit platform. However, the image processing algorithm is highly sensitive to the video's frame rate. GridFree [7] is a Python package for image analysis of interactive grain counting and measurement. It uses K-Means to and principal component analysis (PCA) on both raw image channels and their color indices. It exhibits great performance on multiple crop types. Parico et al. [8] performed real-time pear fruit detection and counting using YOLOv4 models and Deep SORT algorithm. The region-of-interest (ROI) line technique was used to estimate the number of pear fruits detected by the neural network model. Wu et. al. [9] performed detection of Camellia oleifera fruit in complex scenes by using YOLOv7 and data augmentation. The experiment yielded a Mean Average Precision, Precision, Recall, F1 Score, and average detection time of 96.03%, 94.76%, 95.54%, 95.15%, and 0.025 seconds per image respectively.

III. Hardware Components

We propose a low-cost setup for the capture of seed kernel videos for algorithmic analysis using YOLOv8. Fig. 1 shows the seed kernel image capture setup designed for the experiment. The mechanical hopper delivers seeds at a constant rate. The mobile phone is placed on a 3-D printed stand to ensure that the camera is always held orthogonal to the surface to eliminate any skew that may result during the capture of the video. The 3-D printed platform at the bottom channels the seed kernels ensuring that the seed kernels remain in the field of view of the camera as they roll down the lightbox. The mobile phone used for image capture is a Google Pixel 2 XL mobile phone whose default capture frame rate is 60 fps. Fig. 2 shows a frame of the wheat seed kernel video captured using the proposed setup in Fig. 1.

IV. Domain Randomization and Image Datasets

Domain Randomization (DR) trains neural network models on a small sample of images containing simulated objects that

Fig. 1. Mechanical hopper delivering seed kernels

Fig. 2. Wheat seed kernels flowing down the light box

translate closely to real-world objects. Fig. IV shows soy seeds being captured by the proposed image capture setup. Images of 25 seed kernels of soy and wheat are captured using the setup shown in Fig. 3. Using the synthetic image generator developed as part of a previous work [3] synthetic images containing seed kernels of soy and wheat are developed. The synthetic images allow for about 25% overlap at the maximum to account for clustered seed kernels as the frames of the video are processed. Datasets are created for the seed types of soy and wheat, wherein each dataset consists of 200 images of size 320x320x3 with each image containing between 25 and 50 seed kernels overlaid on a light background, as shown in Fig. 4. The generator outputs annotation files that contain location coordinates pertinent to each seed kernel in the image in the TXT format for YOLOv8 to consume and process during training. An additional 35 synthetic images containing 30 seed kernels of each seed kernel type are generated for testing on YOLOv8.
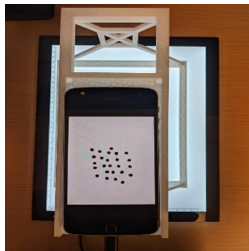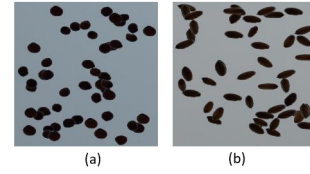

Fig. 3. Image capture of soy seed kernels

Fig. 4. Image capture of soy seed kernels

V. YOLOv8 and Object Tracking Algorithms

The YOLO model is a single-shot detector [10]–[12] that uses a fully convolutional neural network as the backbone to process the input image. The YOLOv8 [13] model was released in January 2023 by Ultralytics. It comprises a convolutional neural network divided into two parts: backbone and head. The backbone, CSPDarknet53 [14], consists of 53 convolutional layers. The head consists of multiple convolutional layers followed by a series of fully connected layers. Object tracking requires that the object be detected in every frame across the video. Several object tracking algorithms [17]–[19] have been proposed over the years. This paper considers two object tracking algorithms for experimentation, namely, StrongSORT and ByteTrack.

A. StrongSORT

The StrongSORT [17] algorithm is an improvement over the DeepSORT [18] algorithm. It is a two-branch framework consisting of an Appearance branch and a Motion branch. The Appearance branch identifies the features of each of the objects detected in a given frame. BoT [20] is leveraged as the feature extractor by the StrongSORT algorithm. The appearance state for the ith tracklet within frame t, e_i^t, is updated as the exponential moving average (EMA) given by $e_i^t = \alpha e_i^{t-1} + (1-\alpha)f_i^t$ where f_i^t is the appearance embedding of the current matched detection and $\alpha = 0.9$, is a variable momentum term. The Motion branch leverages Kalman Filter [23] to predict the position of the object in the frame based on a constant velocity model. The StrongSORT algorithm uses the NSA Kalman Filter algorithm borrowed from the GIAO tracker [24].

B. ByteTrack

ByteTrack [21] algorithm leverages bounding boxes at all confidence levels agnostic of any threshold and attempts to identify all objects in a frame. It uses Tracklets queue to store all the objects (and bounding boxes) that have been detected by the object detector (YOLOv8). The bounding boxes are separated into high score (D^{high}) and low score (D^{low}) based on threshold. Each of the objects in the Tracklets queue is tracked across each frame of the video using Kalman Filter [23]. Firstly, the position of each of the objects in the tracklets queue is predicted in the subsequent frame. The predictions are matched with the actual detections made by the object detector using Motion Similarity score which is computed with Intersection over Union (IoU) between the predicted and actual bounding boxes. Initially, tracklet matching is done between

the predicted and high score (D^{high}) bounding boxes. The tracklets that do not match with any of the high score bounding boxes are matched with low score (D^{low}) bounding boxes. Any tracklet that is not matched is preserved for a predefined number of frames to test for rebirth in case of occlusion. Finally, the tracklet is removed from the queue if a match is never found.

VI. EXPERIMENT

The development of the Seed Counter using YOLOv8 and object tracking algorithms involves three steps: Seed detection, Seed Tracking, and Seed Counting.

A. Seed Detection

The YOLOv8 model is trained on the image dataset (described in section IV) using transfer learning. 80% of the image dataset is used for training and 20% is used for validation. The test dataset consists of 100 images, 50 of soy and 50 of wheat, each containing 20-30 seed kernels. Model weights from the YOLOv8 model pre-trained on the COCO image dataset are leveraged as provided by Uralytics. The hyperparameters used to train the model are as shown in Table I. The results of seed kernel detection on the test dataset are evaluated using the metrics of Precision, Recall, and Average Precision (PR). The metrics are briefly described below and the obtained results are presented in Table II.

TABLE I
HYPERPARAMETERS FOR YOLOV8

Hyperparameter	Value
Learning Rate	0.001
Batch Size	32
Input Image Size	320x320x3 px
Bounding Box Confidence Threshold	0.4
Non-Maximum Suppresion Threshold	0.4
Intersection-over-Union Threshold	0.5
Activation Function	LeakyReLU
Filters in Each Layer	64
Dropout	yes
Pretrained Model Weights	yolov8x

Precision: Precision is given by *true positives/ (true positives + false positives).*
Recall: Recall is given by *true positives/(true positives + false negatives).*
Average Precision: The Average Precision is computed for an 50% overlap between ground truth and predicted bounding boxes for the purposes of this paper, given by AP_{50}.
Note: Average Precision is only reported for the validation data set but not test data set because the images in the test set do not have ground truth bounding boxes plotted around them.
From the results in table II, high recall scores of 91% and 90% on soy and wheat respectively for the test set indicates that the model albeit being trained on synthetic images detects real seed kernels well. The precision scores of 93% and 92% indicate that the model classifies the seed kernels correctly on most instances. The reason for high precision might be due to the clear morphometric distinction between soy and wheat.

TABLE II
EVALUATION RESULTS

Seed Kernel	Validation Set			Test Set	
	Precision	Recall	AP_{50}	Precision	Recall
Soy	98%	92%	92%	93%	91%
Wheat	97%	93%	89%	92%	90%

TABLE III
UNIQUE ID COUNT ON VIDEO OF WHEAT SEED KERNELS BY STRONGSORT

Frame Rate	Seed Kernel Count	Unique IDs
30	250	306
60	250	381
120	250	533

B. Seed Tracking

The Seed Tracking phase applies StrongSORT and Byte-Track algorithms on videos with 250 seed kernels of each seed type captured at three different frame rates, 30, 60, and 120. Both algorithms are applied using the detection weights obtained in the Seed Detection phase as input. The algorithms apply a unique ID to each seed kernel detected in the video and track them throughout the video. The seed kernels in the video are clustered in parts, occluded, and prone to sudden deviations in trajectory, as shown in Fig. 2. These issues lead to the risk of object tracking algorithms assigning different unique IDs to the same seed kernel in different frames of the video, leading to a discrepancy. The number of unique IDs generated by the algorithms on each of the videos captured for the wheat seeds are shown in Tables III and IV respectively. The results show that either of the algorithms consistently overcount the number of seed kernels in the video.

C. Seed Counting

Seed Counting uses a region of interest (RoI) established at a common location across each frame in the video. Any seed kernel that crosses the RoI is accounted to be one seed kernel. The total number of seed kernels is given by the total number of tracks that cross the RoI. Tables V and VI show the results obtained by applying the StrongSORT and ByteTrack algorithms on each of the videos. The performance of the algorithms improves as the frame rate increases. The key issue faced by the algorithms is sudden changes in trajectory due to the seed kernels touching one another and deviating from their present trajectory. This phenomenon affects the object tracking algorithms's ability to predict the location of the seed kernel accurately in subsequent frames.

TABLE IV
UNIQUE ID COUNT ON VIDEO OF WHEAT SEED KERNELS BY BYTETRACK

Frame Rate	Seed Kernel Count	Unique IDs
30	250	322
60	250	406
120	250	592

TABLE V

RESULTS OF SOY AND WHEAT KERNEL COUNT USING STRONGSORT

Seed Type	Frame Rate	Actual Count	YOLOv8 Count	Accuracy
Soy	30	250	238	95.2
Soy	60	250	214	85.6
Soy	120	250	188	75.2
Wheat	30	250	233	93.2
Wheat	60	250	207	82.8
Wheat	120	250	166	66.4

TABLE VI

RESULTS OF SOY AND WHEAT KERNEL COUNT USING BYTETRACK

Seed Type	Frame Rate	Actual Count	YOLOv8 Count	Accuracy
Soy	30	250	242	96.8
Soy	60	250	211	84.4
Soy	120	250	194	77.6
Wheat	30	250	231	92.4
Wheat	60	250	209	83.6
Wheat	120	250	171	68.4

From the results in tables V and VI, the seed count is most accurate on videos captured at a frame rate of 30 and least on videos captured at a frame rate of 120 for either seed type, and object tracking algorithm demonstrating that a lower frame that captures a higher level of detail positively influences the performance of object tracking algorithms. However, both algorithms undercount the number of seed kernels agnostic of frame rate due to the clustering of seed kernels.

VII. PITFALLS, FUTURE WORK, AND CONCLUSION

The key pitfall of the experiment is that the videos used for the experiment consist of seed kernels that are clustered. As a result, the object tracking algorithms failed to track each seed kernel accurately. In further experiments, the video capture mechanism will be altered to ensure that the videos do not contain clustered (or occluded) seed kernels. Overall, the experiment demonstrates the feasibility of synthetic images to train object tracking neural network models, and their application in seed kernel counting aimed at the seed production industry. As the results are encouraging, future work will involve the development of a mobile application (Android/iOS) and a robust video capture mechanism.

REFERENCES

[1] Li, Lei and Zhang, Qin and Huang, Danfeng. A review of imaging techniques for plant phenotyping, Sensors, vol. 14, pp. 20078–20111, 2014

[2] Santos, Luís and Santos, Filipe N and Oliveira, Paulo Moura and Shinde, Pranjali. Deep learning applications in agriculture: A short review, Robot 2019: Fourth Iberian Robotics Conference: Advances in Robotics, Volume 1, pp. 139–151, 2020, Springer

[3] Margapuri, Venkat and Neilsen, Mitchell. Seed phenotyping on neural networks using domain randomization and transfer learning, 2021 ASABE Annual International Virtual Meeting, pp. 1, 2021, American Society of Agricultural and Biological Engineers

[4] Tobin, Josh and Fong, Rachel and Ray, Alex and Schneider, Jonas and Zaremba, Wojciech and Abbeel, Pieter. Domain randomization for transferring deep neural networks from simulation to the real world, 2017 IEEE/RSJ international conference on intelligent robots and systems (IROS), pp. 23–30, 2017, IEEE

[5] Zakharov, Sergey and Kehl, Wadim and Ilic, Slobodan. Deceptionnet: Network-driven domain randomization, Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 532–541, 2019

[6] Neilsen, Mitchell L and Courtney, Chaney and Amaravadi, Siddharth and Xiong, Zhiqiang and Poland, Jesse and Rife, Trevor. A dynamic, real-time algorithm for seed counting, Proc. Of the 26th International Conference on Software Engineering and Data Engineering, 2017

[7] Hu, Yang and Zhang, Zhiwu, GridFree: a Python package of image-analysis for interactive grain counting and measuring. Plant physiology, volume=186, pp. 2239–2252, 2021, Oxford University Press

[8] Real time pear fruit detection and counting using YOLOv4 models and deep SORT, Parico, Addie Ira Borja and Ahamed, Tofael. Sensors, volume=21, pp. 4803, 2021

[9] Wu, Delin and Jiang, Shan and Zhao, Enlong and Liu, Yilin and Zhu, Hongchun and Wang, Weiwei and Wang, Rongyan, Detection of Camellia oleifera fruit in complex scenes by using YOLOv7 and data augmentation, Applied Sciences, volume=12, pp. 11318, 2022, MDPI

[10] Liu, Wei and Anguelov, Dragomir and Erhan, Dumitru and Szegedy, Christian and Reed, Scott and Fu, Cheng-Yang and Berg, Alexander C. Ssd: Single shot multibox detector, Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14, pp. 21–37, 2016

[11] Fu, Cheng-Yang and Liu, Wei and Ranga, Ananth and Tyagi, Ambrish and Berg, Alexander C. Dssd: Deconvolutional single shot detector, arXiv preprint arXiv:1701.06659, 2017

[12] Magalhães, Sandro Augusto and Castro, Luís and Moreira, Germano and Dos Santos, Filipe Neves and Cunha, Mário and Dias, Jorge and Moreira, António Paulo. Evaluating the single-shot multibox detector and YOLO deep learning models for the detection of tomatoes in a greenhouse, Sensors, volume 21, pp. 3569, 2021

[13] Terven, Juan and Cordova-Esparza, Diana. A comprehensive review of YOLO: From YOLOv1 to YOLOv8 and beyond, arXiv preprint arXiv:2304.00501, 2023

[14] Mahasin, Marsa and Dewi, Irma Amelia. Comparison of CSPDarkNet53, CSPResNeXt-50, and EfficientNet-B0 Backbones on YOLO V4 as Object Detector, International Journal of Engineering, Science and Information Technology, volume 2, pp. 64–72, 2022

[15] Ramachandran, Prajit and Parmar, Niki and Vaswani, Ashish and Bello, Irwan and Levskaya, Anselm and Shlens, Jon. Stand-alone self-attention in vision models, Advances in neural information processing systems, volume 32, 2019

[16] Zhao, Hengshuang and Jia, Jiaya and Koltun, Vladlen. Exploring self-attention for image recognition, Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 10076–10085 2020

[17] Du, Yunhao and Zhao, Zhicheng and Song, Yang and Zhao, Yanyun and Su, Fei and Gong, Tao and Meng, Hongying. Strongsort: Make deepsort great again, IEEE Transactions on Multimedia, 2023, IEEE

[18] Dang, Tuan Linh and Nguyen, Gia Tuyen and Cao, Thang. Object tracking using improved deep SORT YOLOv3 architecture, ICIC Express Letters, volume 14, pp. 961–969, 2020

[19] Yolox: Exceeding yolo series in 2021, Ge, Zheng and Liu, Songtao and Wang, Feng and Li, Zeming and Sun, Jian, arXiv preprint arXiv:2107.08430, 2021

[20] A strong baseline and batch normalization neck for deep person re-identification. Luo, Hao and Jiang, Wei and Gu, Youzhi and Liu, Fuxu and Liao, Xingyu and Lai, Shenqi and Gu, Jianyang, IEEE Transactions on Multimedia, volume 22, pp. 2597–2609, 2019, IEEE

[21] Zhang, Yifu and Sun, Peize and Jiang, Yi and Yu, Dongdong and Weng, Fucheng and Yuan, Zehuan and Luo, Ping and Liu, Wenyu and Wang, Xinggang. Bytetrack: Multi-object tracking by associating every detection box, European Conference on Computer Vision, pp. 1–21, 2022, Springer

[22] Girshick, Ross. Fast r-cnn, Proceedings of the IEEE international conference on computer vision, pp. 1440–1448, 2015

[23] Li, Qiang and Li, Ranyang and Ji, Kaifan and Dai, Wei. Kalman filter and its application, 2015 8th International Conference on Intelligent Networks and Intelligent Systems (ICINIS), pp. 74–77, 2015, IEEE

[24] Du, Yunhao and Wan, Junfeng and Zhao, Yanyun and Zhang, Binyu and Tong, Zhihang and Dong, Junhao. Giaotracker: A comprehensive framework for mcmot with global information and optimizing strategies in visdrone 2021, Proceedings of the IEEE/CVF International conference on computer vision, pp. 2809–2819, 2021

Detecting Automated Generated Text with LLMs

Fernando Aguilar-Canto
Computational Cognitive Sciences Laboratory-CIC
Instituto Politécnico Nacional
Mexico City, Mexico
pherjev@gmail.com

Marco Antonio Cardoso-Moreno
Computational Cognitive Sciences Laboratory-CIC
Instituto Politécnico Nacional
Mexico City, Mexico
mcardosom2021@cic.ipn.mx

Diana Jiménez
Computational Cognitive Sciences Laboratory-CIC
Instituto Politécnico Nacional
Mexico City, Mexico
dianaljl.99@gmail.com

Hiram Calvo
Computational Cognitive Sciences Laboratory-CIC
Instituto Politécnico Nacional
Mexico City, Mexico
hcalvo@cic.ipn.mx

Abstract—The development of Large Language Models (LLMs) like GPT-series and BLOOM has revolutionized Artificial Intelligence, yet it has also brought forth challenges in misuse, such as fake content generation and academic cheating. Detecting whether a text is generated by an LLM or written by a human has become imperative. Fine-tuned LLMs have proven to be a promising approach in this regard. In our study, we fine-tuned seven LLMs (BERT, DeBERTa-v3, RoBERTa, XLM-RoBERTa, GPT-2 Medium, GPT-2 Large, GPT-2 XL) to detect text generated by even larger models (GPT-3 and BLOOM) in the AuTexTification task. Among the models, GPT-2 Medium exhibited the best performance in the testing set, achieving an F1-macro score of 0.83272 and an accuracy of 0.83442, surpassing the benchmark's best-known result.

Index Terms—Generated Text Detection, Large Language Models, AuTexTification, GPT-2

I. INTRODUCTION

The rapid advancements in Large Language Models (LLMs) like the GPT series [1]–[4], Pathways Language Model (PaLM) [5], LLaMA [6], and BLOOM [7], have sparked concerns regarding potential misuse [3], [8]–[12]. To address these challenges, a significant task has emerged – detecting Machine Generated Text (MGT), a binary classification problem.

This paper focuses on fine-tuning seven LLMs using the AuTexTification dataset [13], specifically subtask 1 (MGT detection) in English. The results are presented in Section IV, followed by a summary of our findings in Section V.

II. RELATED WORK

Large Language Models (LLMs) have shown effectiveness in detecting content generated by other LLMs. For instance, Uchendu *et al.* [9] employed various models like GROVER, GPT-2, GLTR, BERT, and RoBERTa to detect text produced by GPT-3. Similarly, current generative models like Chat-GPT and LLaMA have been addressed using LLMs such as RoBERTa [14]–[17], DistilBERT [18], LongFormer [19], and OPT-125M [16].

Regarding the AuTextification task, the top-performing model achieved a F1-macro score of 80.91 (confidence interval: 80.4 to 81.38) [13].

III. METHODOLOGY

To tackle this problem, we propose fine-tuning the following LLMs: (1) BERT-cased Large [20], (2) RoBERTa Large [21], (3) XLM-RoBERTa Large [22], (4) DeBERTA Large [23], (5) GPT-2 Medium, (6) Large, and (7) XL [2].

In all cases, default hyperparameters from HuggingFace were used. For GPT-2 re-training, the model served as a feature extractor with a classification layer added, and only one epoch was employed. For BERT-based models, the entire model was fine-tuned using a low learning rate (3e-5) and three epochs. A train-validation split of 80-20 was utilized, and evaluation was performed with the proposed testing set of the task.

IV. RESULTS

The primary outcomes for the BERT-based models are displayed in tables I (validation) and III (testing), while the GPT-2 results are presented in tables II (validation) and IV (testing). Evidently, the most promising comparative results were achieved with GPT-2 models, particularly GPT-2 Medium.

TABLE I
MAIN RESULTS ON THE VALIDATION SET OF THE SUBTASK 1 WITH THE
BERT-BASED MODELS.

Model	1	2	3	4
F1-macro	0.89187	0.90477	0.90647	0.90647
F1-weighted	0.89166	0.90459	0.90633	0.93589
Accuracy	0.89216	0.90501	0.90663	0.93588
Precision	0.82763	0.82763	0.84626	0.90505
Recall	0.90505	0.90505	0.90505	0.90505

V. CONCLUSIONS

This paper conducted a comprehensive comparison of five distinct Large Language Models (LLMs) with seven configurations to address the task of detecting text generated by other LLMs (specifically GPT-3 and BLOOM, [13]). Although there were no significant disparities between BERT-based and GPT-2 models in the validation set, notable differences emerged in the testing set. Notably, GPT-2 Medium exhibited superior

TABLE II
MAIN RESULTS ON THE VALIDATION SET OF THE SUBTASK 1 WITH THE
GPT-BASED MODELS.

Model	5	6	7
F1-macro	0.88900	0.92701	0.93169
F1-weighted	0.88933	0.92704	0.92704
Accuracy	0.89040	0.92703	0.93176
Precision	0.90190	0.92723	0.93178
Recall	0.89040	0.92703	0.93176

TABLE III
MAIN RESULTS ON THE TESTING SET OF THE SUBTASK 1 WITH THE
BERT-BASED MODELS.

Model	1	2	3	4
F1-macro	0.46458	0.49088	0.62265	0.62345
F1-weighted	0.47063	0.49646	0.62597	0.62676
Accuracy	0.57306	0.49646	0.66897	0.66966
Precision	0.54565	0.55434	0.60833	0.60875
Recall	0.99812	0.9990	0.99437	0.99500

TABLE IV
MAIN RESULTS ON THE TESTING SET OF THE SUBTASK 1 WITH THE
GPT-BASED MODELS.

Model	5	6	7
F1-macro	0.83272	0.71571	0.71588
F1-weighted	0.83314	0.71759	0.71776
Accuracy	0.83442	0.73548	0.73571
Precision	0.84168	0.80296	0.80365
Recall	0.83442	0.73548	0.73548

performance, surpassing the F1-macro score of the previous best model and displaying a considerable advantage over the other GPT-2 models.

ACKNOWLEDGMENT

REFERENCES

[1] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever *et al.*, "Improving Language Understanding by Generative Pre-Training," 2018.
[2] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever *et al.*, "Language Models are Unsupervised Multitask Learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.
[3] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, "Language Models are Few-Shot Learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.
[4] OpenAI, "GPT-4 Technical Report," *ArXiv*, vol. abs/2303.08774, 2023, unpublished.
[5] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann *et al.*, "PaLM: Scaling Language Modeling with Pathways," *arXiv preprint arXiv:2204.02311*, 2022, unpublished.
[6] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar *et al.*, "LLaMA: Open and Efficient Foundation Language Models," *arXiv preprint arXiv:2302.13971*, 2023, unpublished.
[7] T. L. Scao, A. Fan, C. Akiki, E. Pavlick, S. Ilić, D. Hesslow, R. Castagné, A. S. Luccioni, F. Yvon, M. Gallé *et al.*, "BLOOM: A 176B-Parameter Open-Access Multilingual Language Model," *arXiv preprint arXiv:2211.05100*, 2022, unpublished.
[8] R. Zellers, A. Holtzman, H. Rashkin, Y. Bisk, A. Farhadi, F. Roesner, and Y. Choi, "Defending Against Neural Fake News," *Advances in Neural Information Processing Systems*, vol. 32, 2019.
[9] A. Uchendu, T. Le, K. Shu, and D. Lee, "Authorship Attribution for Neural Text Generation," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020, pp. 8384–8395.
[10] M. Weiss, "Deepfake Bot Submissions to Federal Public Comment Websites Cannot Be Distinguished from Human Submissions," *Technology Science*, vol. 2019121801, 2019.
[11] L. Weidinger, J. Mellor, M. Rauh, C. Griffin, J. Uesato, P.-S. Huang, M. Cheng, M. Glaese, B. Balle, A. Kasirzadeh *et al.*, "Ethical and social risks of harm from language models," *arXiv preprint arXiv:2112.04359*, 2021, unpublished.
[12] D. Yan, M. Fauss, J. Hao, and W. Cui, "Detection of AI-generated Essays in Writing Assessment," *Psychological Testing and Assessment Modeling*, vol. 65, no. 2, pp. 125–144, 2023.
[13] A. M. Sarvazyan, J. Á. González, M. Franco Salvador, F. Rangel, B. Chulvi, and P. Rosso, "Overview of AuTexTification at IberLEF 2023: Detection and Attribution of Machine-Generated Text in Multiple Domains," in *Procesamiento del Lenguaje Natural*, Jaén, Spain, sep 2023, in press.
[14] B. Guo, X. Zhang, Z. Wang, M. Jiang, J. Nie, Y. Ding, J. Yue, and Y. Wu, "How Close is ChatGPT to Human Experts? Comparison Corpus, Evaluation, and Detection," *arXiv preprint arXiv:2301.07597*, 2023, unpublished.
[15] Y. Ma, J. Liu, F. Yi, Q. Cheng, Y. Huang, W. Lu, and X. Liu, "AI vs. Human–Differentiation Analysis of Scientific Content Generation," *arXiv preprint arXiv:1911.00650*, 2023, unpublished.
[16] F. Mireshghallah, J. Mattern, S. Gao, R. Shokri, and T. Berg-Kirkpatrick, "Smaller Language Models are Better Black-box Machine-Generated Text Detectors," *arXiv preprint arXiv:2305.09859*, 2023, unpublished.
[17] H. Zhan, X. He, Q. Xu, Y. Wu, and P. Stenetorp, "G3Detector: General GPT-Generated Text Detector," *arXiv preprint arXiv:2305.12680*, 2023, unpublished.
[18] S. Mitrović, D. Andreoletti, and O. Ayoub, "ChatGPT or Human? Detect and Explain. Explaining Decisions of Machine Learning Model for Detecting Short ChatGPT-generated Text," *arXiv preprint arXiv:2301.13852*, 2023, unpublished.
[19] Y. Li, Q. Li, L. Cui, W. Bi, L. Wang, L. Yang, S. Shi, and Y. Zhang, "Deepfake Text Detection in the Wild," *arXiv preprint arXiv:2305.13242*, 2023, unpublished.
[20] J. Devlin, M.-W. 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, Volume 1*, 2019, pp. 4171–4186.
[21] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "RoBERTa: A Robustly Optimized BERT Pretraining Approach," *arXiv preprint arXiv:1907.11692*, 2019, unpublished.
[22] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, É. Grave, M. Ott, L. Zettlemoyer, and V. Stoyanov, "Unsupervised Cross-lingual Representation Learning at Scale," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 8440–8451, unpublished.
[23] P. He, X. Liu, J. Gao, and W. Chen, "DeBERTa: Decoding-enhanced BERT with Disentangled Attention," *arXiv preprint arXiv:2006.03654*, 2020, unpublished.

Explainable Image Recognition with Graph-based Feature Extraction and Classification

Basim Azam
IIIS, Griffith University
Brisbane, Australia
basimazam0@gmail.com

Brijesh Verma
IIIS, Griffith University
Brisbane, Australia
b.verma@griffith.edu.au

Deepthi Praveenlal Kuttichira
IIIS, Griffith University
Brisbane, Australia
d.kuttichira@griffith.edu.au

Abstract— Deep learning models have proven remarkably adept at extracting salient features from raw data, driving state-of-the-art performance across diverse tasks. However, these models suffer from a lack of interpretability; they function as black boxes, obscuring the feature-level underpinnings of their predictions. Addressing this problem, our work presents an innovative framework that fuses the power of convolutional layers for feature extraction with the versatility of Graph Neural Networks (GNNs) to model relationships among neuron activations. Our framework operates in two phases: first, it identifies class-oriented neuron activations by analyzing image features, then these activations are encapsulated within a graph structure. The GNN leverages the relationships among these neuron activations to generate a final, interpretable classification. The proposed model not only matches, but at times exceeds, the accuracy of current leading models, all the while providing transparency via class-specific feature importance. This novel integration of convolutional and graph neural networks offers a significant step towards interpretable and accountable deep learning models.

Keywords— Graph Neural Networks, Convolutional Neural Network

I. INTRODUCTION

The field of image classification has seen significant advancements in recent years, largely due to the advent of Deep Learning (DL) techniques. A critical aspect of these techniques is the extraction of robust and discriminative features from raw image data, which forms the basis for any successful image classification model. These features, which can range from simple color and texture information to complex patterns and objects, provide the necessary input for the model to learn and make accurate predictions.

Feature extraction and image classification have been extensively studied in the field of machine learning and computer vision. Traditional methods for feature extraction include Histogram of Oriented Gradients (HOG), Scale-Invariant Feature Transform (SIFT), and Speeded Up Robust Features (SURF) [1]. These methods, however, require manual engineering and are often not optimal for complex tasks such as image classification [2].

In recent years, CNNs have emerged as a powerful tool for image classification. CNNs are capable of automatically learning hierarchical feature representations from raw pixel data, which has led to significant improvements in image classification performance [3]. Convolutional layers extract features from the input raw data. The classifier component of the CNN architecture maps the features to output class labels. However, despite their effectiveness, CNNs are often criticized for their lack of interpretability. The internal workings of these models are often described as a "black box," making it difficult to understand how they arrive at their predictions [4].

CNNs have been widely adopted in the field of image classification due to their ability to automatically learn hierarchical feature representations from raw pixel data [5]. Despite their success, one of the main criticisms of CNNs is their lack of interpretability [6]. This lack of transparency can be problematic in certain applications where interpretability is crucial, such as medical imaging and autonomous driving [7].

Explainable AI (XAI) has emerged as a promising research direction to address the interpretability issue of deep learning models. XAI aims to make the decision-making process of AI models transparent and understandable to humans [8]. Various methods have been proposed to provide explainability , including saliency maps, layer-wise relevance propagation, deep Taylor decomposition [9], and Graph Neural Networks (GNNs) [10]. Towards this end, we aim to explore graph constructions from neuron activations to provide interpretability of features till the classification output [11]. A potential limitation is the difficulty of handling dynamic graphs, where the structure of the graph changes over time [20].

The research makes significant contributions to the field of computer vision in several notable ways:

- We introduce a novel framework that integrates the power of convolutional layers for feature extraction and robustness of GNNs to model relationship between neuron activations. The framework leads to an advanced model that provides a new level of interpretability.

- Unlike traditional deep learning methods, the proposed architecture provides explainability by tracing predictions back to the specific neurons that contributed to them.

- The proposed model transforms neuron activations into graph structures, which is significant to capture and expose the relationship between different features in an interpretable manner.

- A comprehensive comparison is presented with state-of-the-art models. The proposed model's better accuracy coupled with explainability, and transparency leverages the application of model in decision-making processes.

In the following sections, we will delve into the details of our proposed architecture, the experimental setup, and the results obtained, further demonstrating the effectiveness and advantages of our approach.
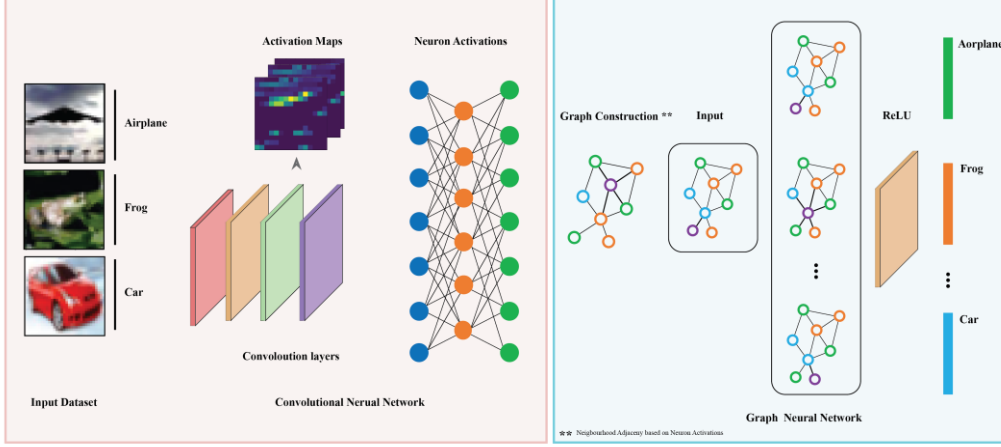
Figure 1 An overview of the proposed architecture providing explanations at each step using graph neural networks.

II. PROPOSED METHODOLOGY

The proposed method for feature extraction and image classification involves the use of Graph Neural Networks (GNNs). GNNs are capable of capturing the complex relationships between nodes in a graph, making them suitable for tasks that involve structured data [16]. In the context of image classification, each pixel in an image can be considered as a node in a graph, and the relationships between pixels can be modelled as edges in the graph. By applying GNNs to this graph representation of an image, we can effectively extract features that capture the spatial relationships between pixels, which can then be used for image classification [17].

The proposed architecture aims to leverage the strengths of both convolutional operations for local image structure extraction and graph-based operations for global relational reasoning. The overall model involves a feature extraction component, a graph construction component, a graph convolution component, and a classification component. These components are not isolated but interconnected, forming an end-to-end trainable system.

Given an input image $x_i \in \mathbb{R}^{H \times W \times C}$, where H, W, and C are height, width, and number of channels of the image, respectively. We utilize the deep neural network, which we denote as $F(\cdot)$, to transform the input image into a set of features defined by the mapping function F_i, where each feature corresponds to the output of a specific filter. This transformation can be represented as :

$$F_i = F(x_i; \theta_F)$$

Where $F_i \in \mathbb{R}^{H' \times W' \times C'}$ are the height, width, and number of channels of the feature maps, θ_F denotes the parameters for the feature extraction.

The next step is to construct a graph $G_i = V_i, A_i$ from feature maps F_i. Each nod $v_{i,j} \in V_i$ in the graph corresponds to a region of the image and is assigned a feature vector $f_{i,j}$ extracted from F_i. The edges of the graph represent the relationships between different regions of the image. The adjacency matric $A_i \in \mathbb{R}^{n \times n}$, where n is the number of nodes, is defined based on the relationship between the feature vectors of the nodes as f the nodes as

$$A_{i,j,k} = \frac{f_{i,j} \cdot f_{i,k}}{\|f_{i,j}\|_2 \|f_{i,k}\|_2}$$

where $A_{i,j,k}$ id the entry at the jth row and kth column of A_i, and $\cdot$ denotes the dot product.

We then perform graph convolution operation, denoted as $GC(.)$, to propagate information through the graph. The operation updates the node features based on their own features and the features of their neighbours capturing the relational information between different regions of the image. The node embedding N_i^l, after l graph convolution layers can be represented as:

$$N_i^l = \sigma(GC(N_i^{(l-1)} A_i; \theta_{GC}^l))$$

where $N_i^l \in \mathbb{R}^{n \times d_l}$ denotes the node embeddings after l layers, $N_i^o = V_i$, θ_{GC}^l denotes the parameters of the lth graph convolution layer, d_l is the dimension of the node embeddings after l layers, and $\sigma(\cdot)$ and is a non-linear activation function, such as the ReLU function.

The node embeddings N_i^L after L layers of graph convolution are then aggregated to generate a graph embedding $G_i \in \mathbb{R}^{d_L}$. This is done using the softmax function:

$$G_i' = softmax\left(\frac{1}{n}\sum_{j=1}^{n} N_{i,j}^L\right)$$

where $N_{i,j}^L$ is the jth row of N_i^L

The graph embedding G_i' is then passed through a classifier to predict the output label $y_{i,hat}$:

$$y_{i,hat} = argmax(C'(G_i'; \theta_C))$$

where θ_C denotes the parameters of the classifier, and the $argmax(\cdot)$ returns the index of the maximum value, indicating the predicted class.

The model is trained by optimizing the parameters $\theta = \{\theta_F, \theta_{GC}^l, \dots, \theta_{GC}^L, \theta_C\}$ to minimize the negative likelihood loss between the predicted labels and the true labels, which can be given as:

$$L(\theta) = -\frac{1}{N}\sum_{i=1}^{N}\log\left(y_{i,hat}\left[y_i\right]\right)$$

where N is the number of samples, $y_{i,hat}[y_i]$, denotes the predicted probability of the true class for the ith sample, and the log is the natural logarithm.

III. EXPERIMENTAL SETUP

In this section, the description of dataset, evaluation metrics and details of hardware and software setup used for the experiments are discussed.

MNIST [26], CIFAR-10 [27], and Kaggle Cats and Dogs [28] dataset are used in this study. The evaluation metrics used to evaluate the proposed methodology are accuracy, precision, recall, and f1-score.

Our model achieved an accuracy of 99.26% on the MNIST dataset as indicated in Table 1, indicating that it correctly classified almost every image of the test set. The precision and recall were 99.26%, suggesting that our model was effective in identifying the correct digits and minimizing false positives and negatives. The F1 score, a measure of the model's overall performance, was 99.2% demonstrating a good balance between precision and recall.

Table 2 presents the performance of our model on the CIFAR-10 dataset. This dataset is more complex than MNIST, containing colour images of 10 different classes, including animals and vehicles. On the CIFAR-10 dataset, our model achieved an accuracy of 0.8737, meaning it correctly classified 87.37% of the images. The precision and recall were 87.35% and 87.37%, respectively. These values indicate that our model was effective in correctly identifying the classes of the images while minimizing false positives and negatives. The F1 score was 87.31%, indicating a balanced performance between precision and recall. Table 3 represent the scores achieved Kaggle cats and dogs dataset for each individual category separately, and overall

The input images are processed by the convolution layers to extract the highly important features. The model deems these features necessary for the classification. The fully connected layers describes the high-level features computed initially. The fully connected layers are base point to build graph structures, once the activations are further narrowed down, the proposed network transforms these activations into a graph structure. Each graph node corresponds to the neuron

Table 1 Class-wise performance evaluation on MNIST dataset.

Class/Metric	Precision	Recall	F1-Score
0	0.97	0.99	0.98
1	0.99	0.99	0.99
2	0.96	0.94	0.95
3	0.97	0.99	0.98
4	0.99	0.99	0.99
5	0.96	0.94	0.95
6	0.97	0.99	0.98
7	0.99	0.99	0.99
8	0.96	0.94	0.95
9	0.97	0.99	0.98
Overall	0.99	0.99	0.99
Accuracy		0.993	

Table 2 Class-wise performance evaluation on CIFAR10 dataset.

Class/Metric	Precision	Recall	F1-Score
Airplane	0.91	0.89	0.90
Automobile	0.93	0.91	0.92
Bird	0.87	0.89	0.88
Cat	0.91	0.89	0.9
Deer	0.93	0.91	0.92
Dog	0.87	0.89	0.88
Frog	0.91	0.89	0.90
Horse	0.93	0.91	0.92
Ship	0.87	0.89	0.88
Truck	0.91	0.89	0.90
Overall	0.87	0.87	0.87
Accuracy		0.87	

Table 3 Class-wise performance evaluation on Kaggle Cats and Dogs dataset

Class/Metric	Precision	Recall	F1-Score
Cat	0.91	0.89	0.90
Dog	0.93	0.91	0.92
Overall	0.91	0.91	0.91
Accuracy		0.91	

Table 4 Performance comparison of the proposed model with State-of-the-art approaches on MNIST and CIFAR-10 datasets.

Model Name	Proposed Model	DNN5 [20]	FastSNN [23]	Tsetlin Machine [22]	Park et. al. [21]	CBoF & DSH [22]	Proposed Model	DNN5 [20]	CBoF & DSH [22]	CCN[26]	ResNet -8 [20]	CVPR_Class [24]
Dataset	MNIST	MNIST	MNIST	MNIST	MNIST	MNIST	CIFAR10	CIFAR10	CIFAR10	CIFAR10	CIFAR10	CIFAR10
Accuracy	99.26	97.2	97.97	98.20	98.10	99.45	87.37	86.5	88.70	83.36	86.5	86.65

in dense layer and edges represent the relationship between neurons.

In Table 4, we present a comprehensive comparison of our proposed model with the state-of-the-art models (DNN5 [20], Fast SNN [23], Tsetlin Machine [22], Park et. al [21], and CCN [25]) on two popular datasets: MNIST and CIFAR10. The performance of each model is evaluated based on accuracy scores. The datasets consist of Airplane, Automobile, Bird, Cat, Deer, Dog, Frog, Horse, Ship, and Truck. The in terms of accuracy scores are as 87.4%, 95.0%, 79.2%, 75.9%m 88.3%, 77.8%, 92.8%, 93.3%, and 91.8%. Figure 3 presents the confusion matrices computed on the MNIST and CIFAR-10 datasets. It can be interpreted from the results that almost all of the classes in both datasets have been classified accurately.

The components of our model include the Convolutional Neural Network (CNN), Neuron Activations, Graph Construction, and Graph Neural Network (GNN). For each class, we evaluated the model's performance with different combinations of these components. The performance metrics used in this study are Accuracy, Precision, Recall, and F1 Score. The scores achieved describe the effectiveness of proposed model along with explainability of the models.

IV. Conclusion

The paper presents a novel explainable graph neural network based approach to produce accurate classification labels. The notable contribution of the approach is to provide explanations to the features and build graph relationships based on neuron activations for each specific class. The proposed graph neural network component of the architecture then models the intricate relationship between the neuron activations. The architecture achieves overall accuracy of 99.26%, 91.44% and 87.37% on MNIST, Cat and Dog Dataset and CIFAR-10 datasets respectively. In comparison to the state-of-the-art approaches, the proposed architecture not only achieves better performance but provided explainability to the features, neuron activations, and the relationship between these activations. The experiments are continued to broaden the applicability of proposed approach, and to evaluate on more diverse image datasets.

V. Acknowledgements

References

[1] Dalal, N., & Triggs, B. (2005). Histograms of oriented gradients for human detection. In 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05) (Vol. 1, pp. 886-893). IEEE.

[2] Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. International journal of computer vision, 60(2), 91-110.

[3] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems (pp. 1097-1105).

[4] Zeiler, M. D., & Fergus, R. (2014, September). Visualizing and understanding convolutional networks. In European conference on computer vision (pp. 818-833). Springer, Cham

[5] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. nature, 521(7553), 436-444.

[6] Z. J. Wang, R. Turko, O. Shaikh, H. Park, N. Das, F. Hohman, M. Kahng, andD. H. P. Chau, "CNN explainer: Learning convolutional neural networks with interactive visualization," *IEEE Transactions on Visualization and Computer Graphics*, vol. 27, no. 2, pp. 1396–1406, 2020.

[7] Holzinger, A., Biemann, C., Pattichis, C. S., & Kell, D. B. (2017). What do we need to build explainable AI systems for the medical domain?. arXiv preprint arXiv:1712.09923.

[8] Montavon, G., Samek, W., & Müller, K. R. (2018). Methods for interpreting and understanding deep neural networks. Digital Signal Processing, 73, 1-15.

[9] Montavon, G., Samek, W., & Müller, K. R. (2018). Methods for interpreting and understanding deep neural networks. Digital Signal Processing, 73, 1-15.

[10] A. Holzinger, B. Malle, A. Saranti, and B. Pfeifer, "Towards multi-modal causability with Graph Neural Networks enabling information fusion for explainable AI," in Information Fusion, vol. 71, pp. 28-37, 2021.

[11] R. G. KDDLab et al., "A survey of methods for explaining Black Box Models," ACM Computing Surveys, vol. 51, pp. 1-42, 2018.

[12] A. Adadi and M. Berrada, "Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI)," in IEEE Access, vol. 6, pp. 52138-52160, 2018.

[13] Zhou, J., Cui, G., Zhang, Z., Yang, C., Liu, Z., Wang, L., ... & Sun, M. (2018). Graph neural networks: A review of methods and applications. arXiv preprint arXiv:1812.08434.

[14] Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., & Yu, P. S. (2020). A comprehensive survey on graph neural networks. IEEE Transactions on Neural Networks and Learning Systems.

[15] Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., & Bengio, Y. (2018). Graph attention networks. arXiv preprint arXiv:1710.10903..10903.

[16] 13. Zhou, J., Cui, G., Zhang, Z., Yang, C., Liu, Z., Wang, L., ... & Sun, M. (2018). Graph neural networks: A review of methods and applications. arXiv preprint arXiv:1812.08434.

[17] Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., & Yu, P. S. (2020). A comprehensive survey on graph neural networks. IEEE Transactions on Neural Networks and Learning System.

[18] 15. Liu, Z., Zhou, J., & Li, L. (2018). GeniePath: Graph Neural Networks with Adaptive Receptive Paths. arXiv preprint arXiv:1802.00910.

[19] Cui, P., Wang, X., Pei, J., & Zhu, W. (2018). A survey on network embedding. IEEE Transactions on Knowledge and Data Engineering, 31(5), 833-852.

[20] Pishchik, E. "Trainable Activations for Image Classification." Preprints.org 2023, 2023010463. doi: 10.20944/preprints202301.0463.v1.

[21] J. Park, J. Lee, and D. Jeon, "7.6 A 65nm 236.5 nJ/classification neuromorphic processor with 7.5% energy overhead on-chip learning using direct spike-only feedback," in 2019 IEEE International Solid-State Circuits Conference-(ISSCC), IEEE, 2019.

[22] N. Passalis and A. Tefas, "Training lightweight deep convolutional neural networks using bag-of-features pooling," IEEE Transactions on Neural Networks and Learning Systems, vol. 30, no. 6, pp. 1705-1715, 2018.

[23] Taylor, Luke, Andrew J. King, and Nicol Spencer Harper. "Robust and accelerated single-spike spiking neural network training with applicability to challenging temporal tasks." (2022).

[24] Wightman, Ross, Hugo Touvron, and Hervé Jégou. "Resnet strikes back: An improved training procedure in timm." arXiv preprint arXiv:2110.00476, October 2021.

[25] Jeevan, Pranav, and Amit Sethi. "Vision Xformers: Efficient attention for image classification." arXiv preprint arXiv:2107.02239, July 2

[26] Li Deng, "The MNIST Database of Handwritten Digit Images for Machine Learning Research ," IEEE Signal Processing Magazine, vol. 29, no. 6, pp. 141–142, Nov. 2012

[27] A. Krizhevsky, "Learning Multiple Layers of Features from Tiny Images," 2009.

[28] O. M. Parkhi, A. Vedaldi, A. Zisserman, and C. V. Jawahar, "Cats and dogs," *IEEE Xplore*, Jun. 01, 2012.