Workshop de Computação Aplicada (WorCAP 2022)

Deep Learning: Transference and Explainability

14 September 2022

Valdivino Alexandre de Santiago Júnior

Coordenação de Pesquisa Aplicada e Desenvolvimento Tecnológico (COPDT)
Instituto Nacional de Pesquisas Espaciais (INPE)
São José dos Campos, SP, Brazil

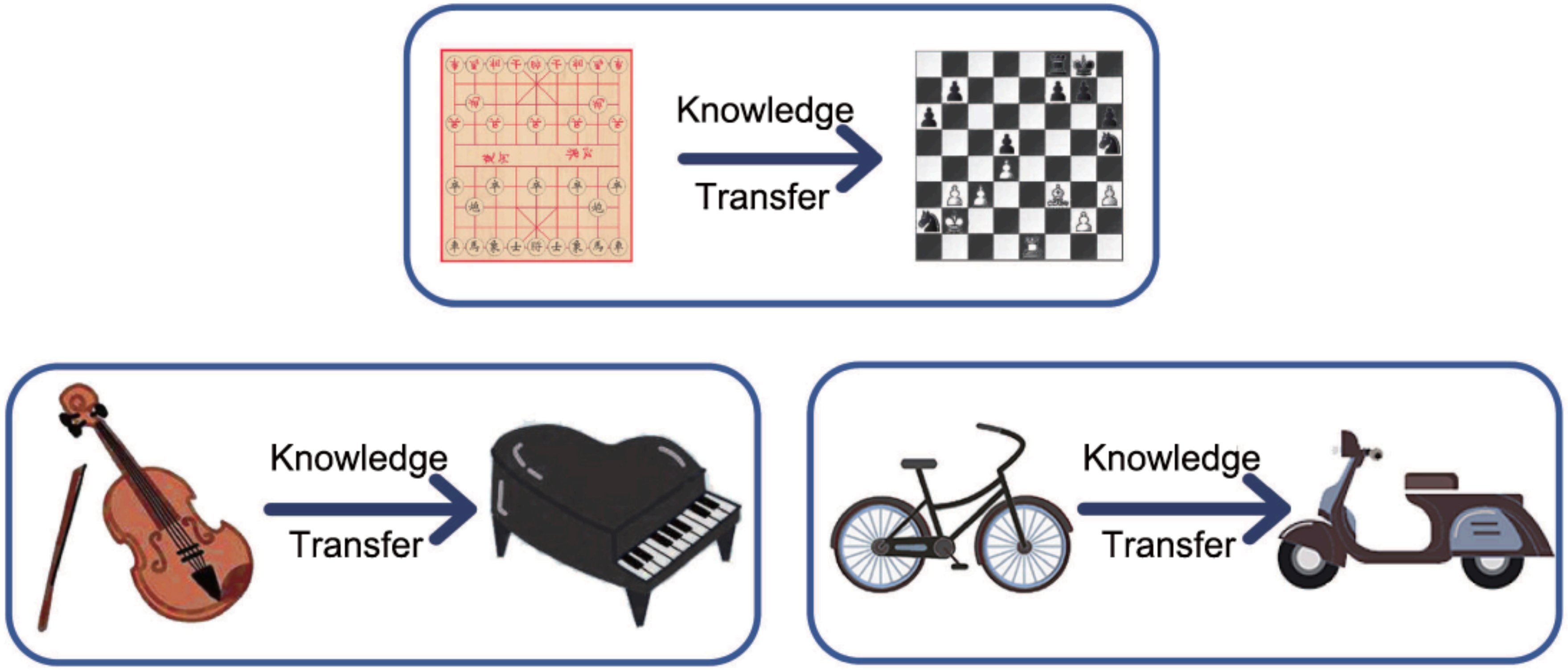
Classroom

Transfer Learning!

Photo by CDC on Unsplash

Transfer Learning

- Improve the performance of **target learners** on **target domains** by transferring the knowledge contained in **related source domains**.



Transfer Learning

- Homogeneous Transfer Learning: $\mathcal{X}_{SOURCE} = \mathcal{X}_{TARGET}$
- Heterogeneous Transfer Learning: $\mathcal{X}_{SOURCE} \neq \mathcal{X}_{TARGET}$

Source: K. Weiss, T. M. Khoshgoftaar, and D. Wang. 2016. A survey of transfer learning. Journal of Big Data 3 (2016), 9.

Medicine (Melanoma)

Satellite

Project IDeepS

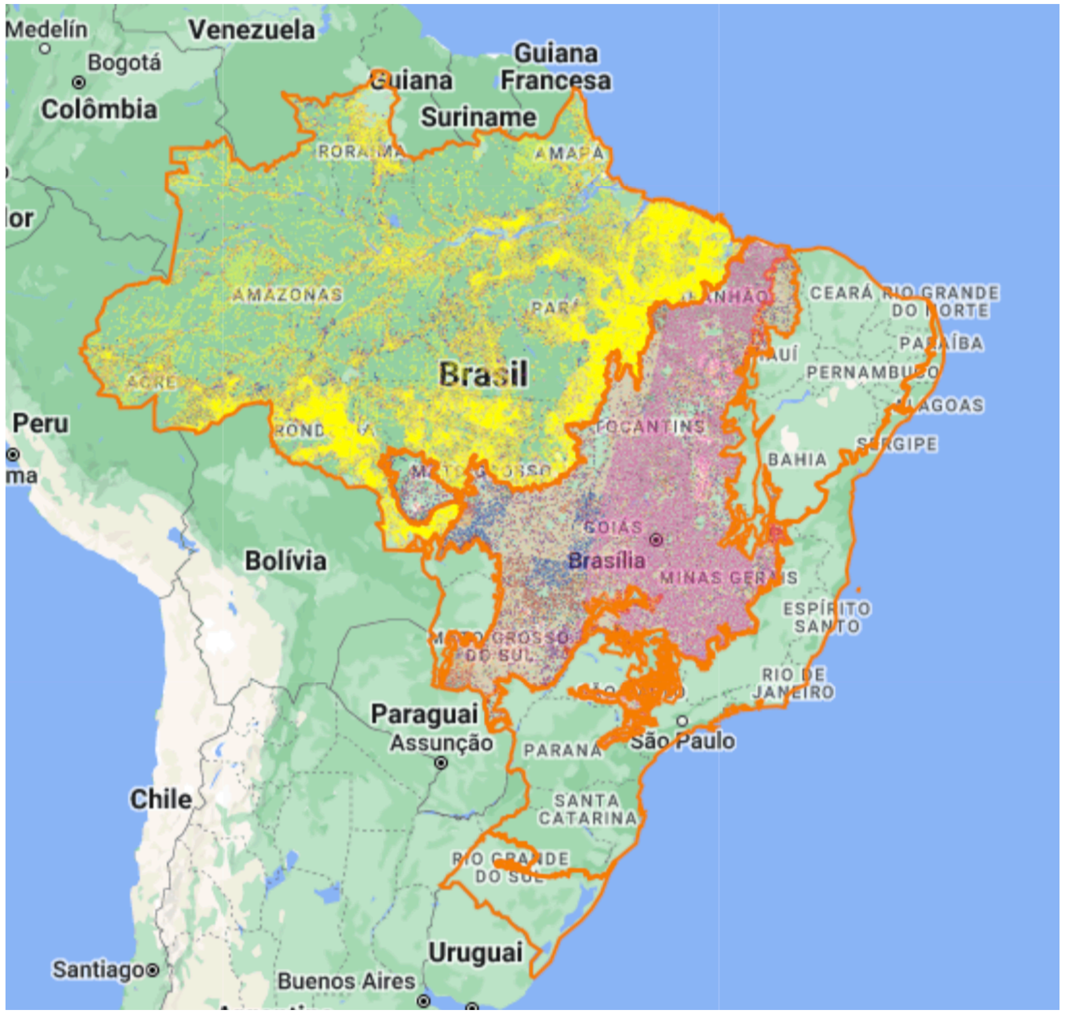
- Classificação de imagens via redes neurais profundas e grandes bases de dados para aplicações aeroespaciais.

Project IDeepS

Source: https://github.com/vsantjr/IDeepS

IDeepS: Objective 1

- Large-scale investigation, deep neural networks (DNNs), satellite image classification.



IDeepS: Objective 2

- Best DNNs, drones, autonomy.

IDeepS: Higher Objective

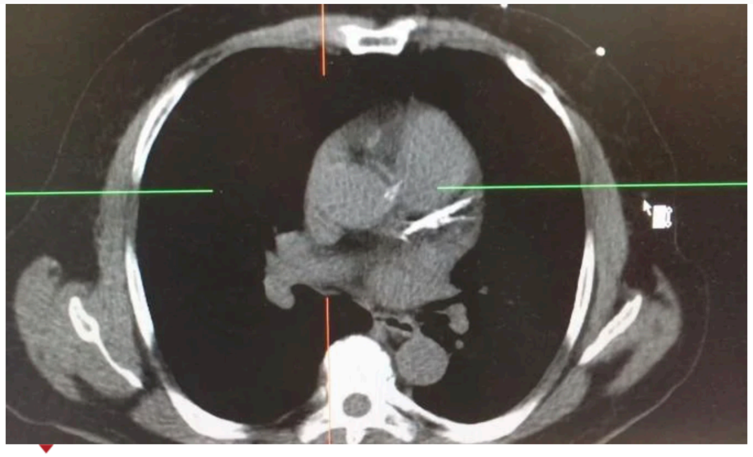
Recommendations/Suggestions

Remote Sensing

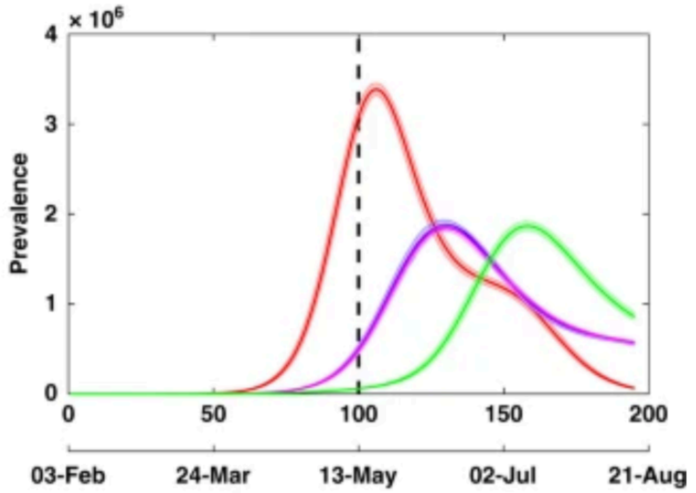
Drones

Scientific Software Testing

- Scientific software: non-trivial outputs such as 2D, 3D.
- Testing is not straightforward: non-deterministic behaviour, non-trivial outputs, test automation (**oracle**).



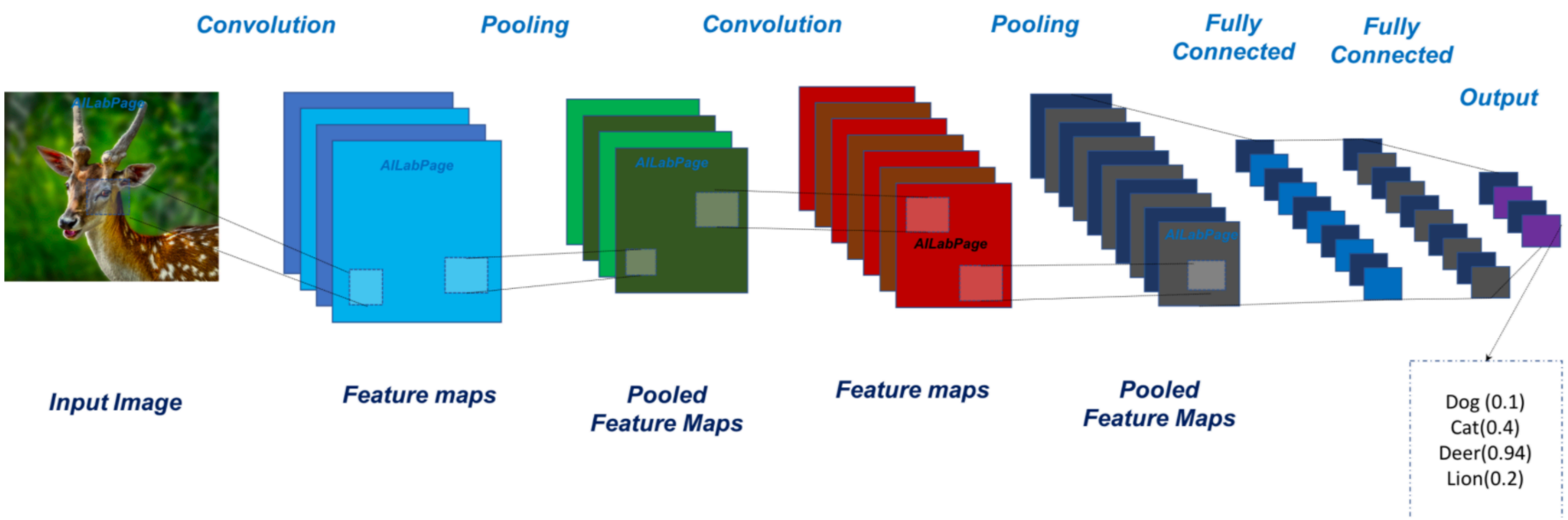
Medicine Software
(CT scan)



Social / Biological Modelling
(COVID-19)

Motivation

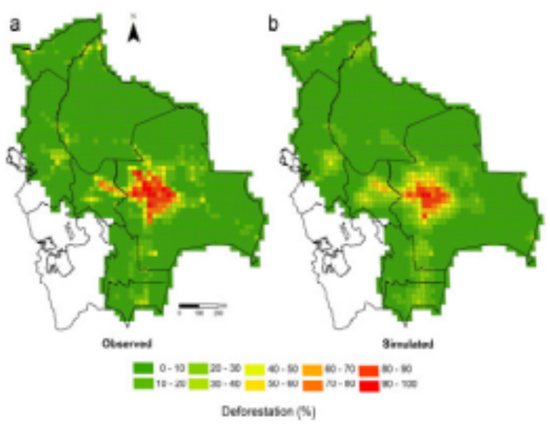
- Deep convolutional neural network (CNN).

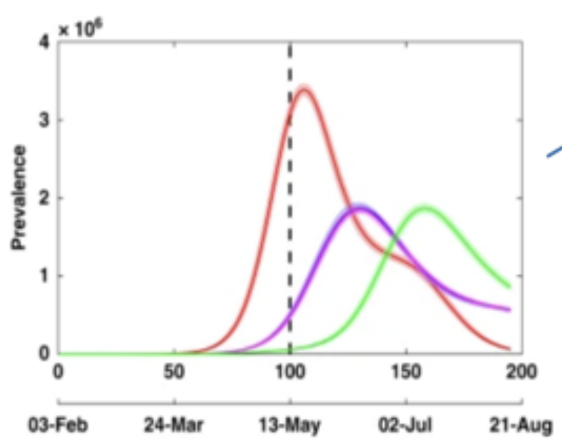


Source: https://vinodsblog.com/2018/10/15/everything-you-need-to-know-about-convolutional-neural-networks/

Motivation

Outputs



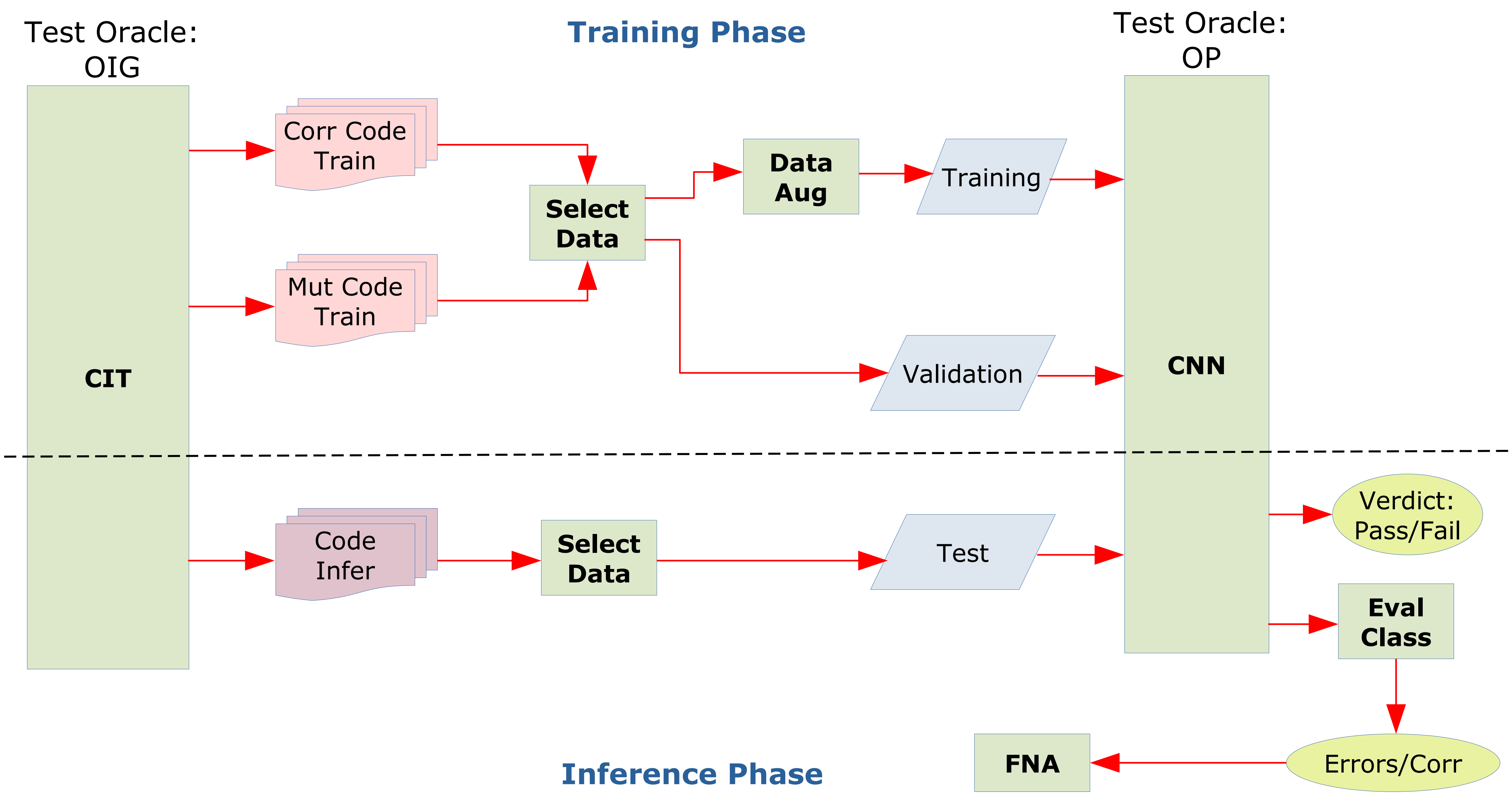


Test Oracle Procedure
(CNN)

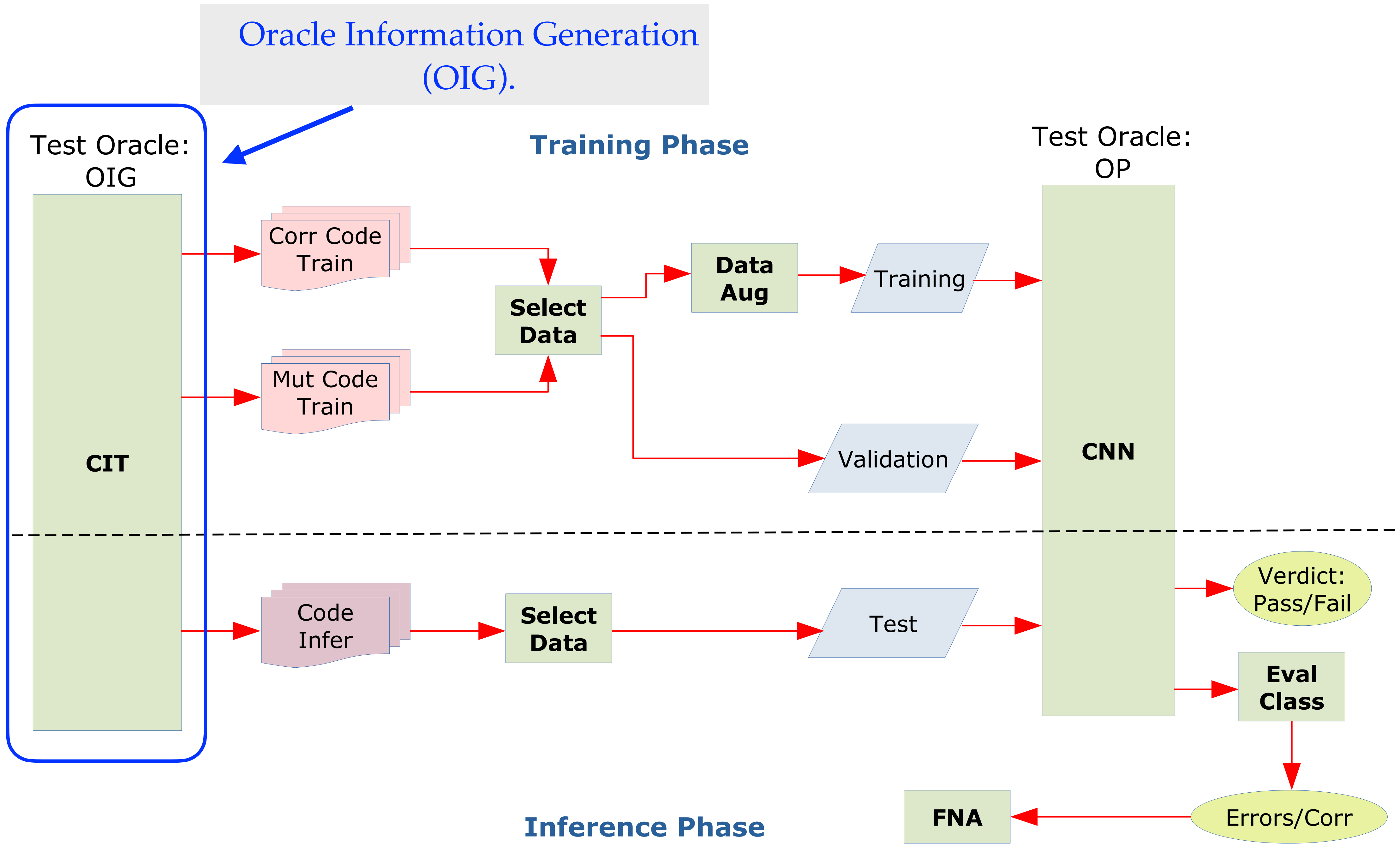
This Study: Main Contributions

- Method: **Test Oracle based on CNN** (**TOrC**).
- Technique: **Feature and Neighbourhood-based Analysis** (**FNA**).

The TOrC Method



TOrC: OIG



CIT = Combinatorial interaction testing.

TOrC: Generating Images

PS: Binary classification problem.

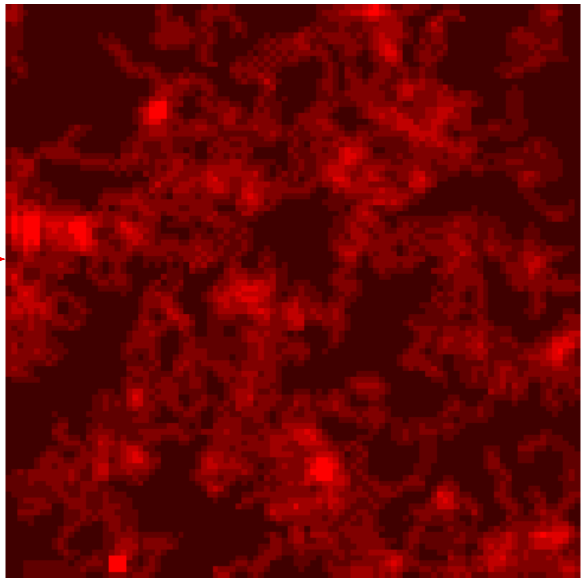
CIT's tuple

0.005
45
0.9
0.9

Scientific Model

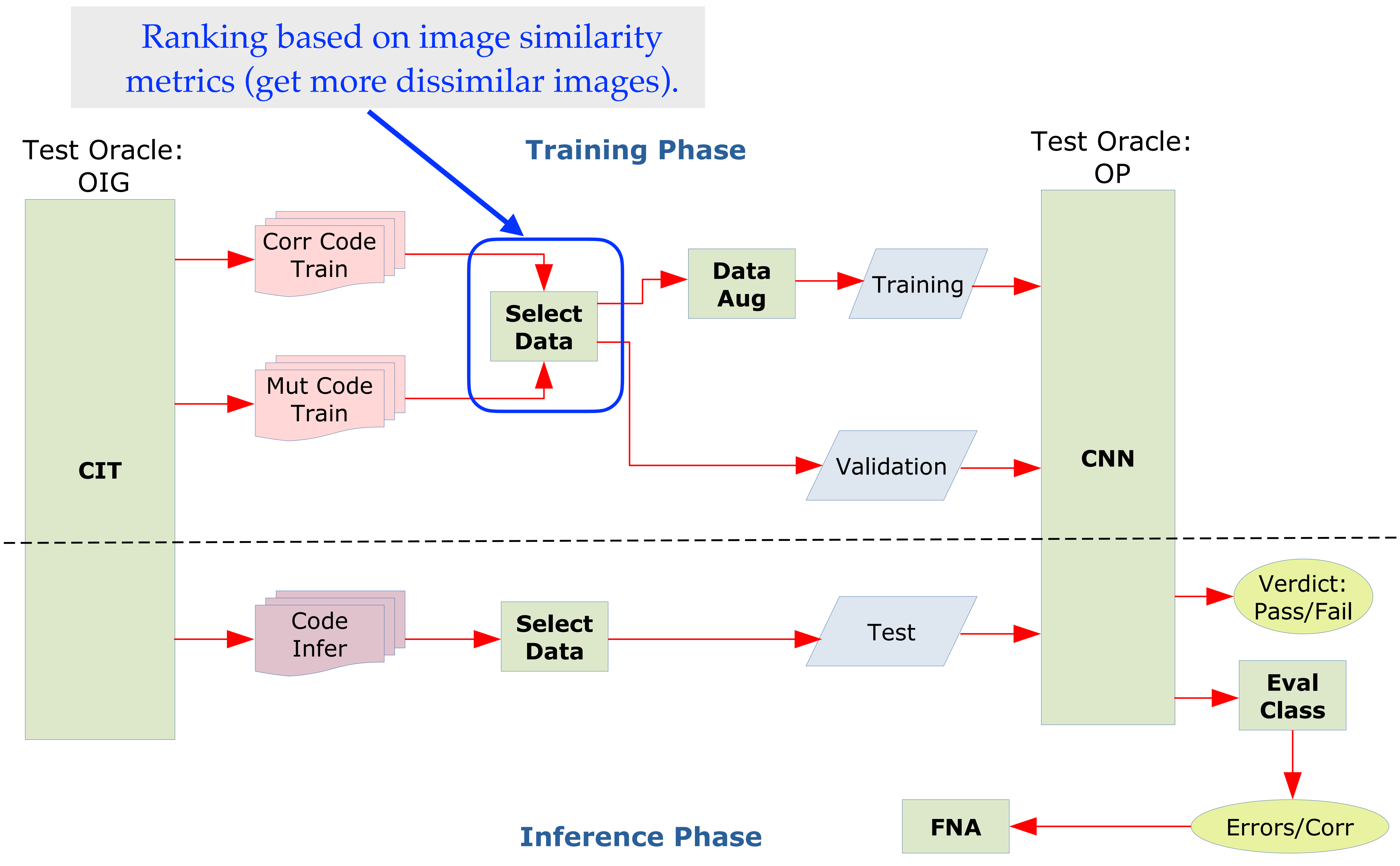
```
Heatbugs = Model{
    quantity = 150,
    dim = 100,
    finalTime = 20,
    evaporationRate = 0.04,
    initialCellTemperature = 10,
    temperature = {
        max = 150,
        min = 0
    },
    idealTemperature = {
        min = 45,
        max = 60
    },
```

Image



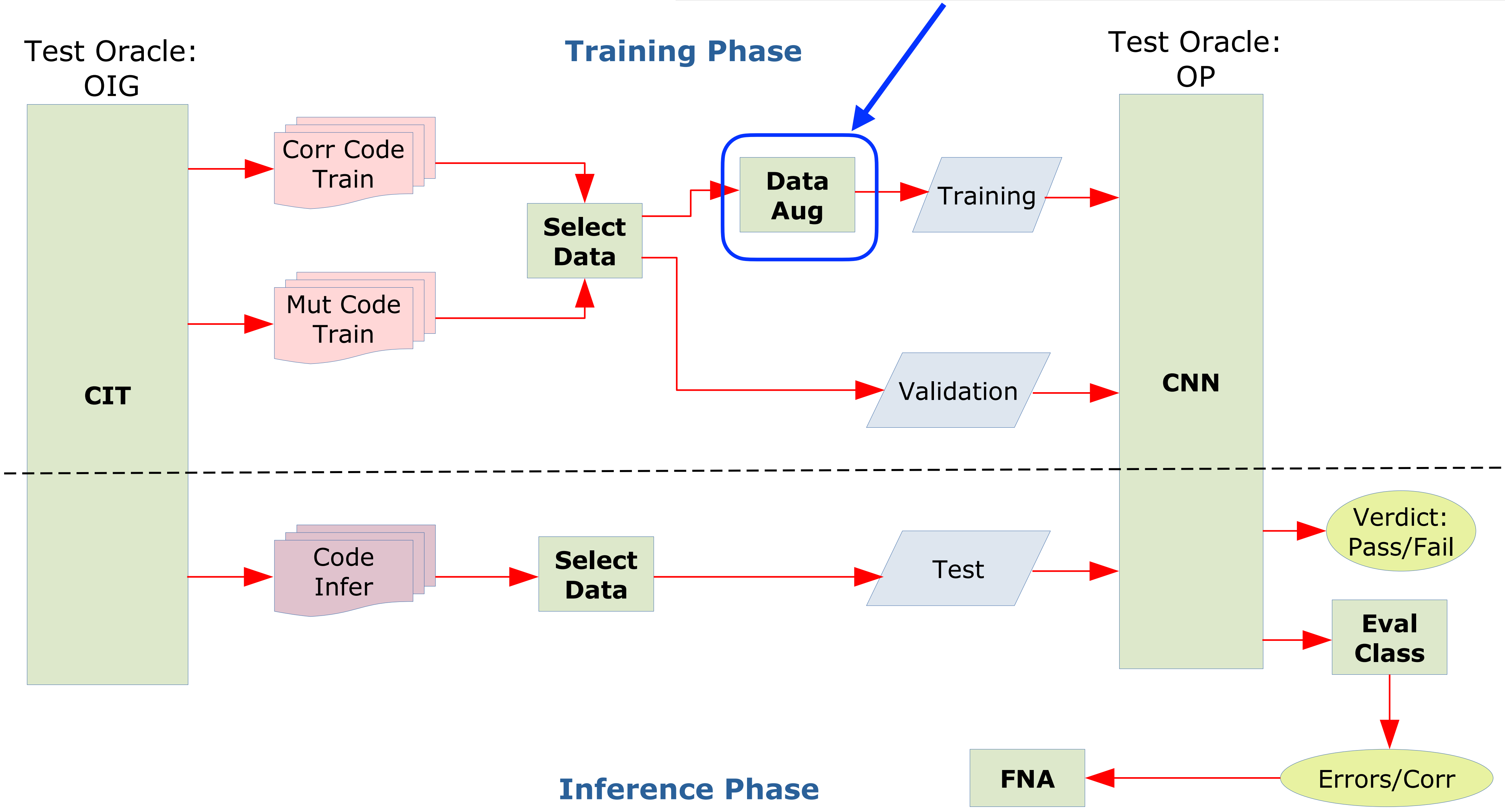
Correct codes (correct class) and
second-order mutants (mutant class).

TOrC: Select Data

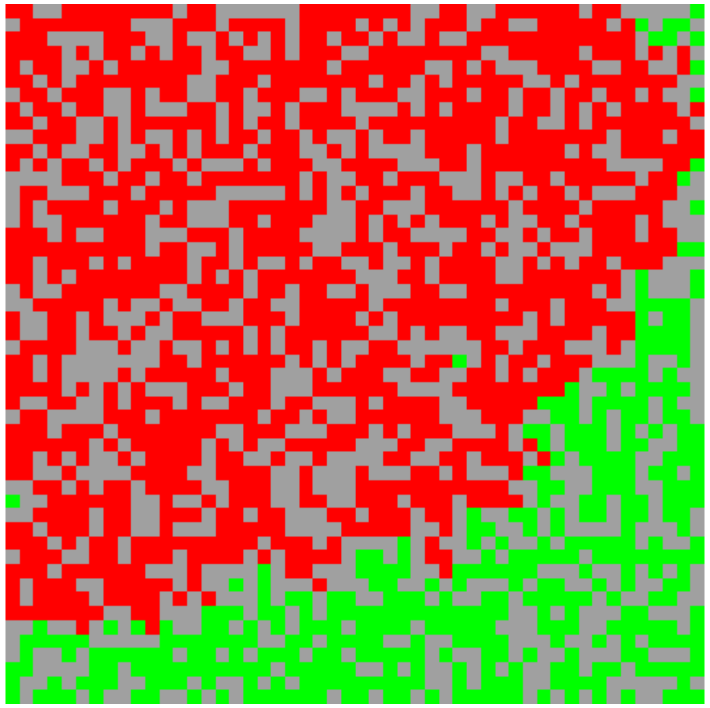


TOrC: Data Augmentation

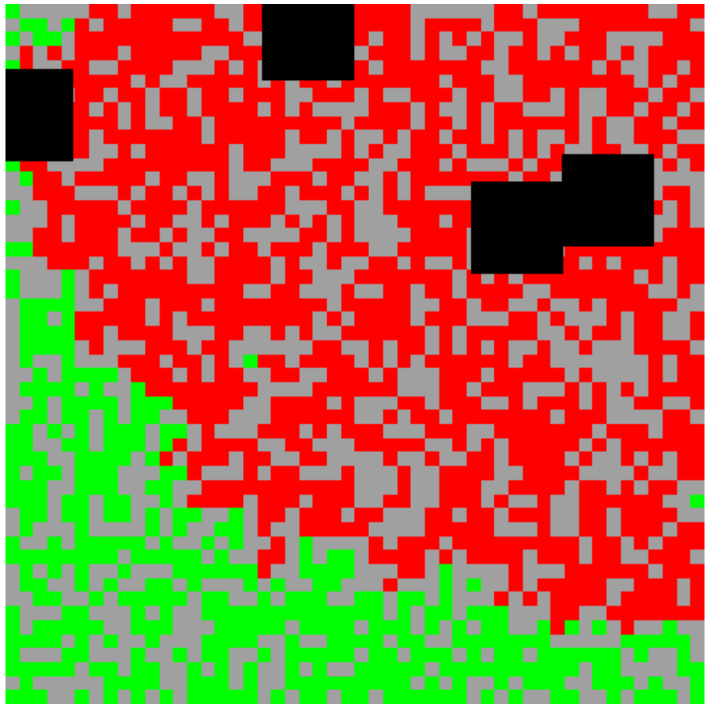
Decreasing the errors (image misclassifications) due to the ML models by reducing overfitting.



TOrC: Data Augmentation



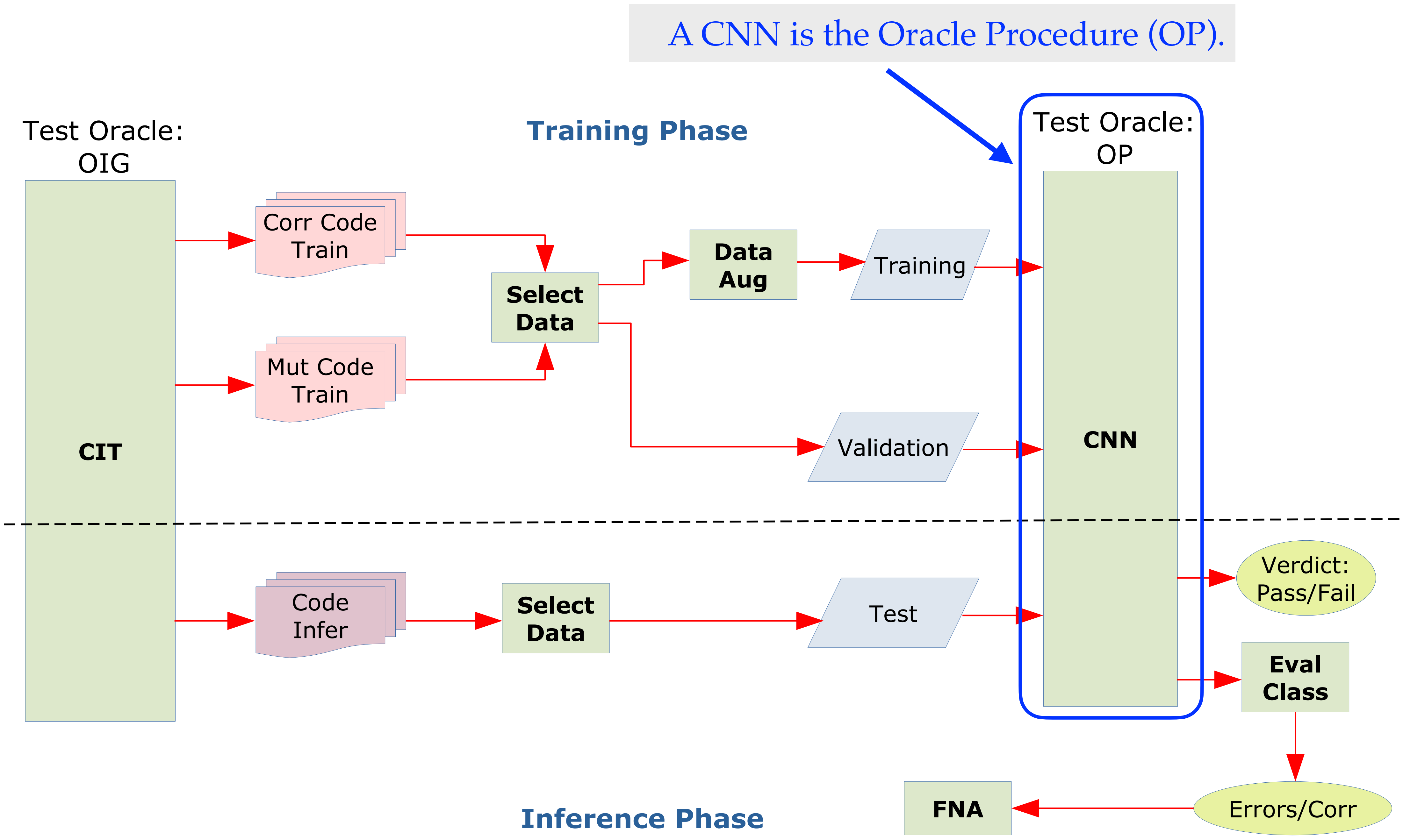
Original image



Data-augmented image
(horizontal flip + cutout transformations)

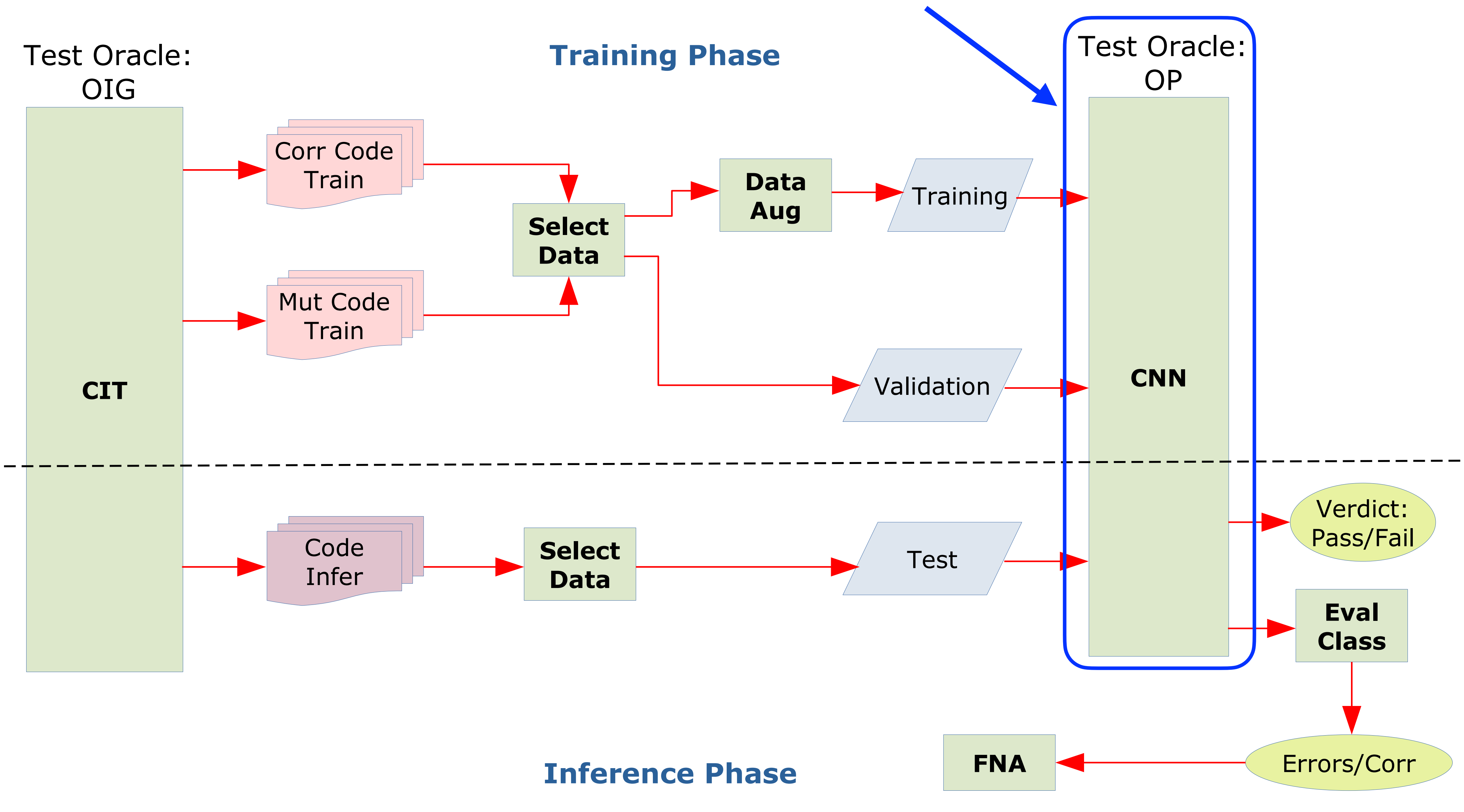
PS: Fire spreading model (cellular space).

TOrC: Oracle Procedure



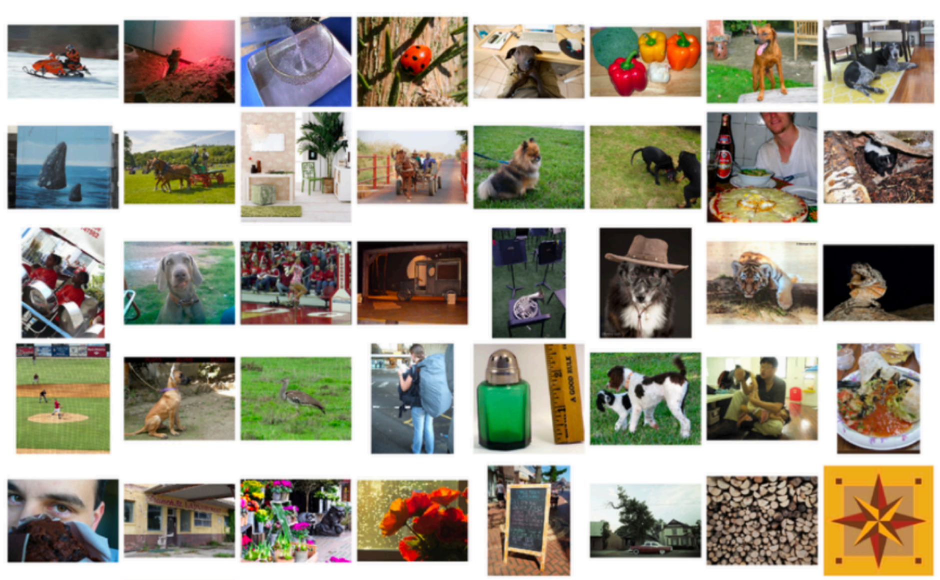
TOrC: Oracle Procedure

Transfer learning (pretrained models).

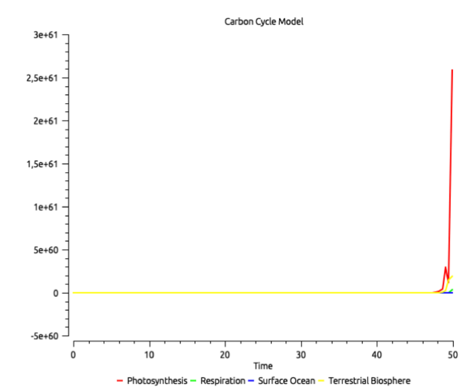


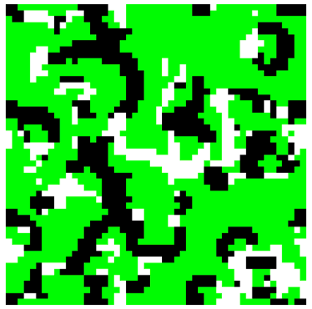
TOrC: Transfer Learning

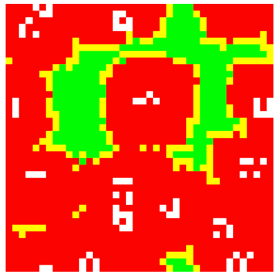
- Fine-tuning: Instead of random initialisation, the model is initialised with a pretrained model. Layers: **unfrozen**.

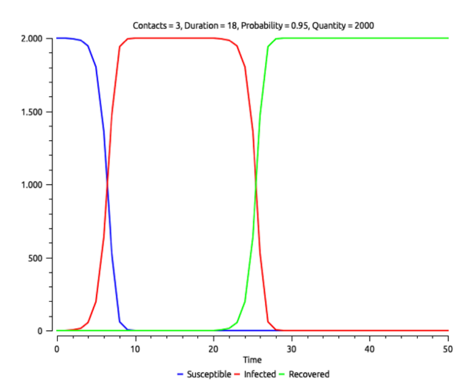


ImageNet





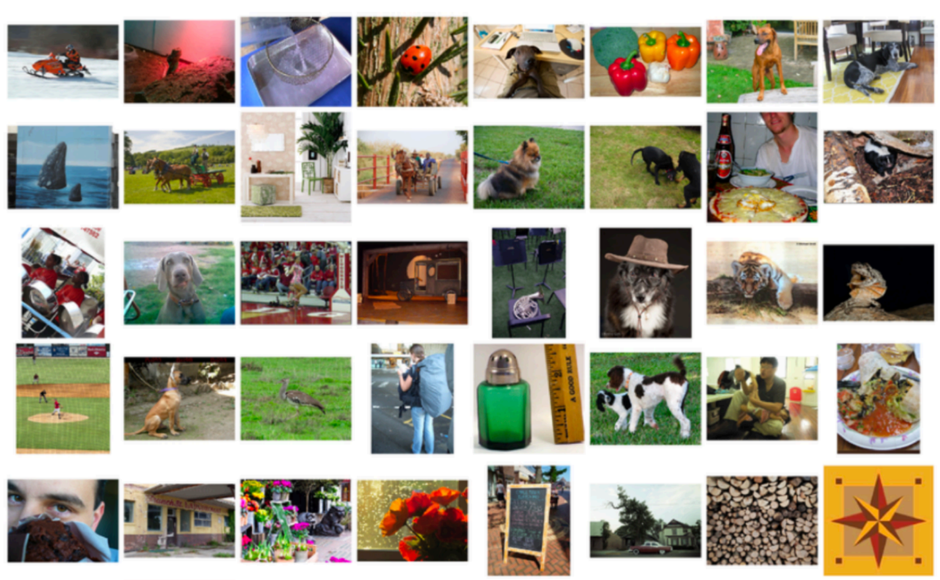




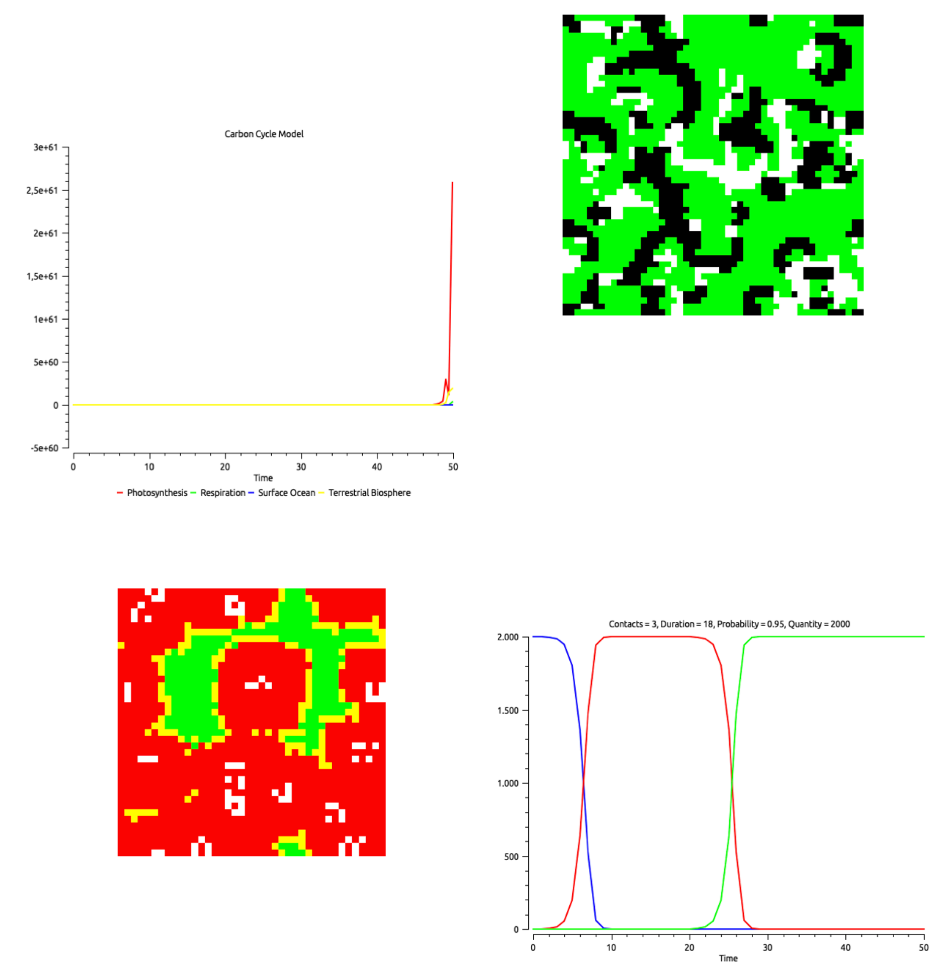
TerraME

TOrC: Transfer Learning

- Fine-tuning and Heterogenous Transfer Learning.

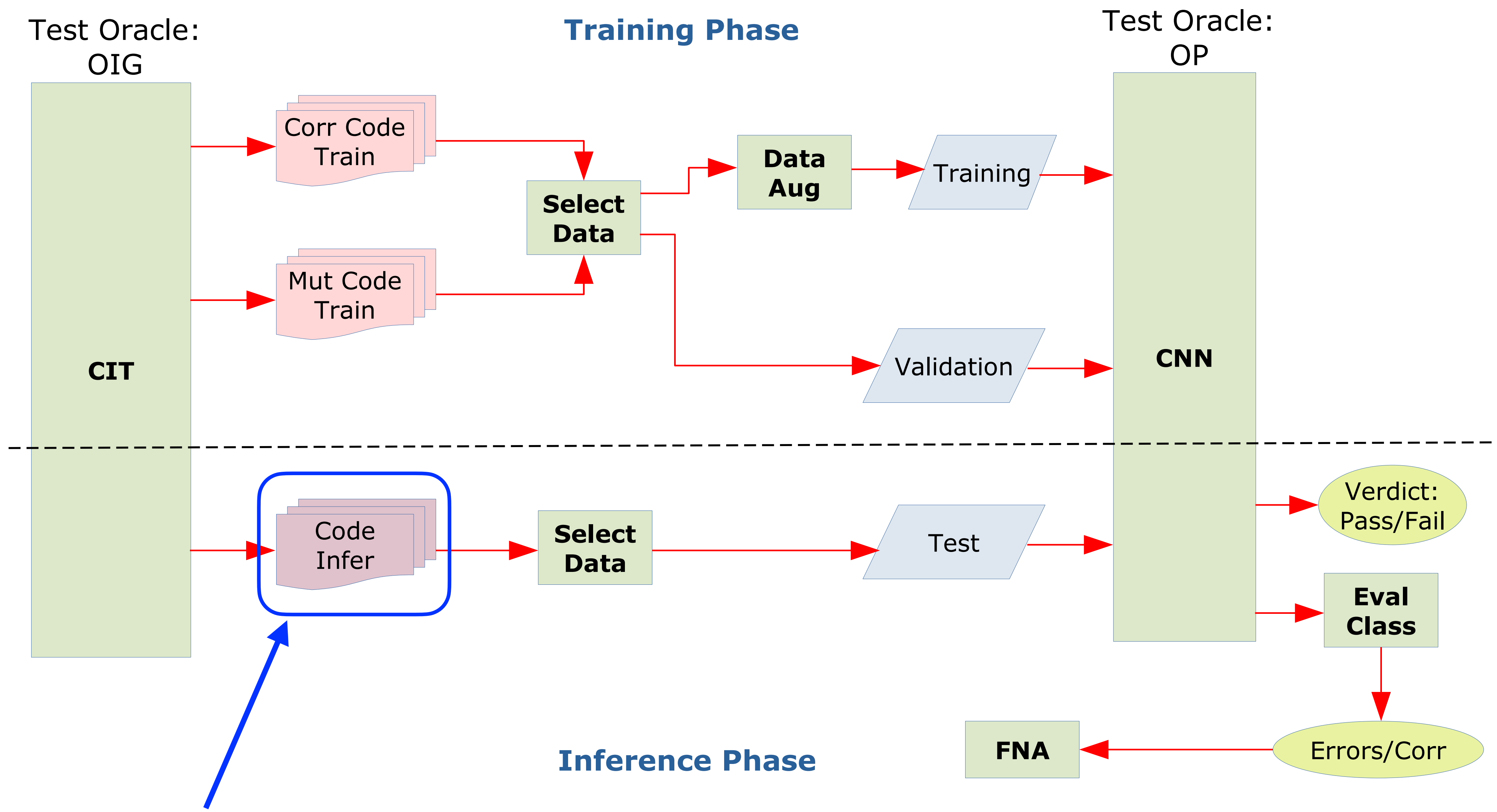


ImageNet



TerraME

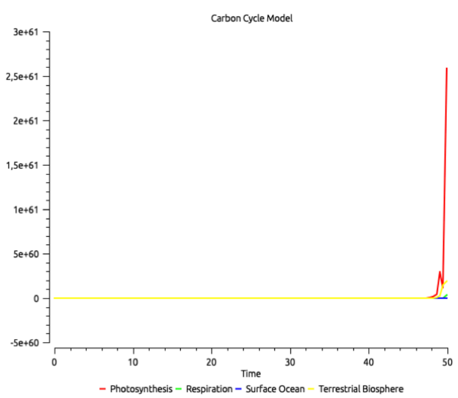
TOrC: Inference Phase

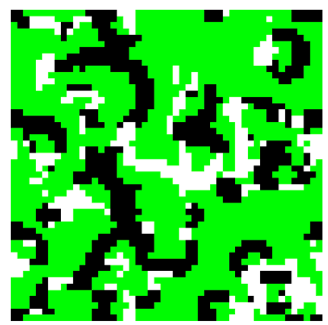


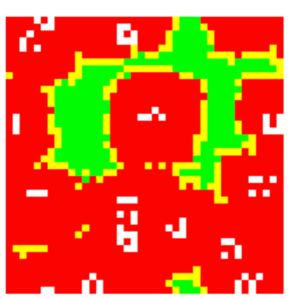
Test set is created based on the outputs of programs completely different from the ones used to create the training and validation sets.

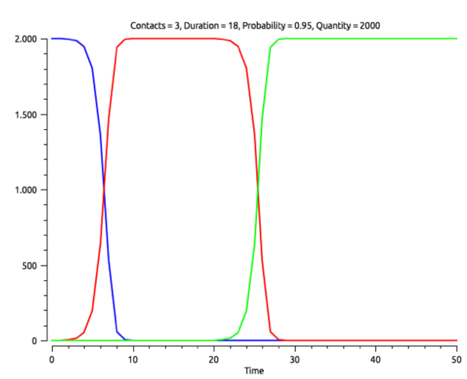
TOrC: Transfer Learning

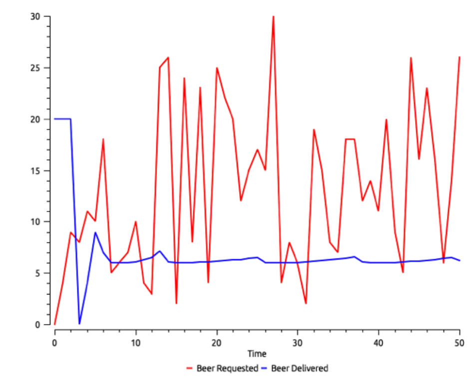
- It is possible that we have a third domain?

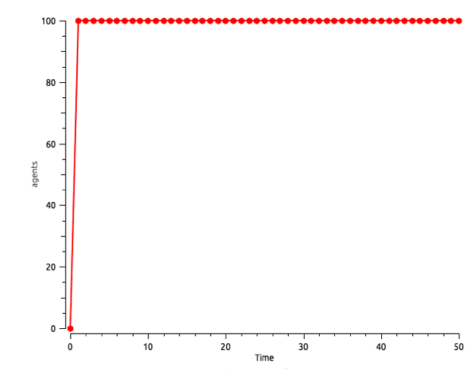


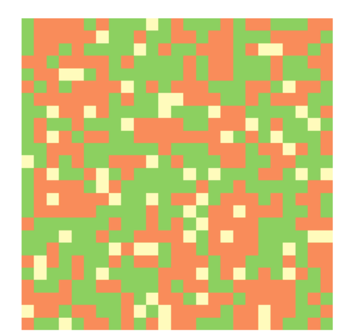












Training Set

Test Set

Experimental Design

- Research Question 1 (**RQ_1**):
  - Does a deeper CNN (more layers) always have better performance compared to a shallower (less layers) one?

- Research Question 2 (**RQ_2**):
  - If we do not change the architecture of a predefined model/ network, is **pure** transfer learning able to get the same or better performances compared to extended architectures of the model?

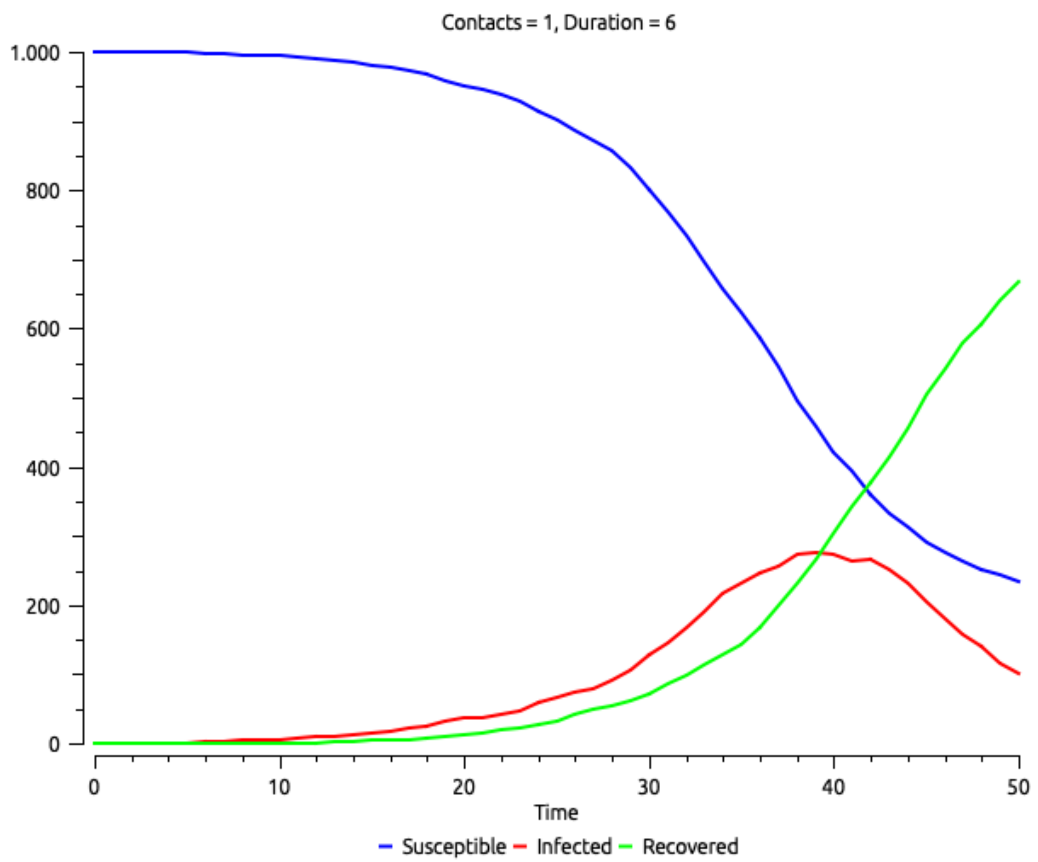
Scientific Models

- Second-order mutants.

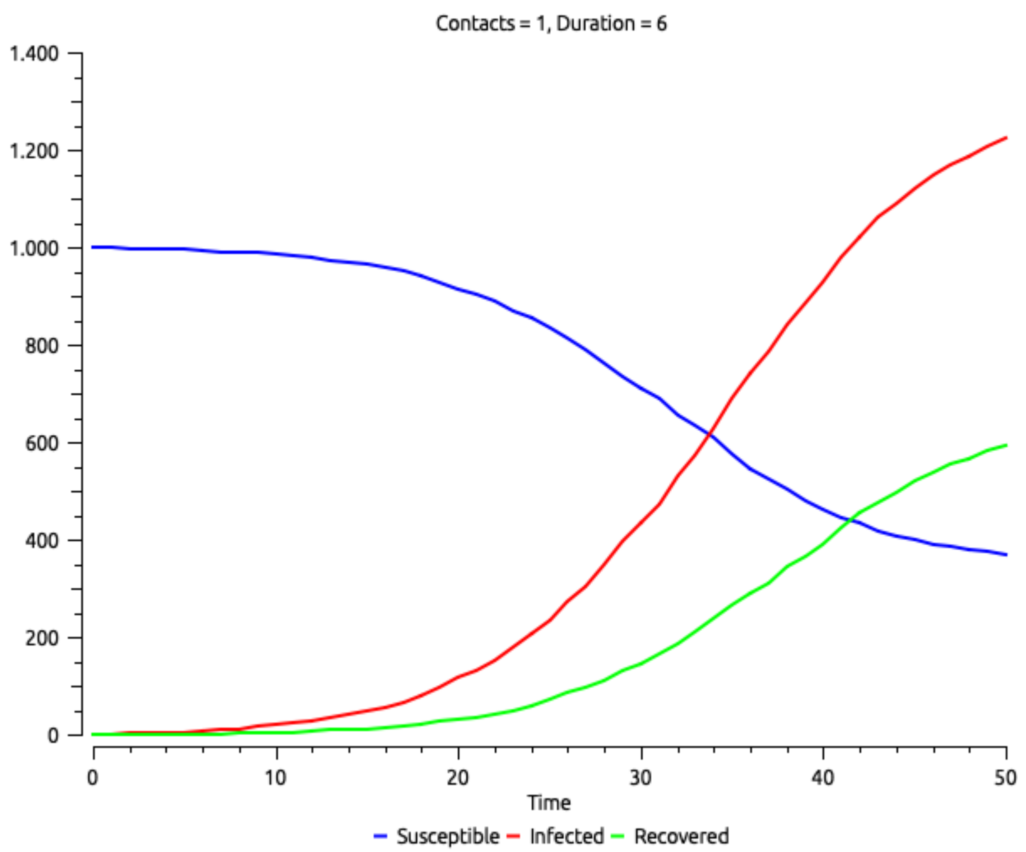
```
if self.state == "infected" then
    forEachConnection(self, function(conn)
        self:message{receiver = conn, delay = 1}
    end)

    -- Mutation 1: ROR4
    -- if self.counter > model.duration then ... CORRECT CODE
    if self.counter == model.duration then
        self.state = "recovered"
        -- Mutation 2: AOR1
        -- model.infected = model.infected - 1 ... CORRECT CODE
        model.infected = model.infected + 1
        model.recovered = model.recovered + 1
    end
```

Samples



SIR model: correct



SIR model: mutant

PS: Susceptible, Infected and Recovered (SIR) model (plot).
COVID-19.

CNNs

CNN	#Layers	#TL	#TLE1L	#TLE2L	#In Feat
ResNet-18 [20]	18	11.17M	11.44M	11.83M	512
ResNet-34 [20]	34	21.28M	21.55M	21.94M	512
ResNeXt-50-32x4d [62]	50	22.98M	27.18M	27.96M	2,048
Wide ResNet-50-2 [64]	50	66.83M	71.03M	71.81M	2,048
Inception v3 [52]	48	21.78M	25.98M	26.76M	2,048
ResNet-152 [20]	152	58.14M	62.34M	63.12M	2,048
DenseNet-161 [23]	161	26.47M	31.35M	32.17M	2,208

CNNs

Architecture configurations.

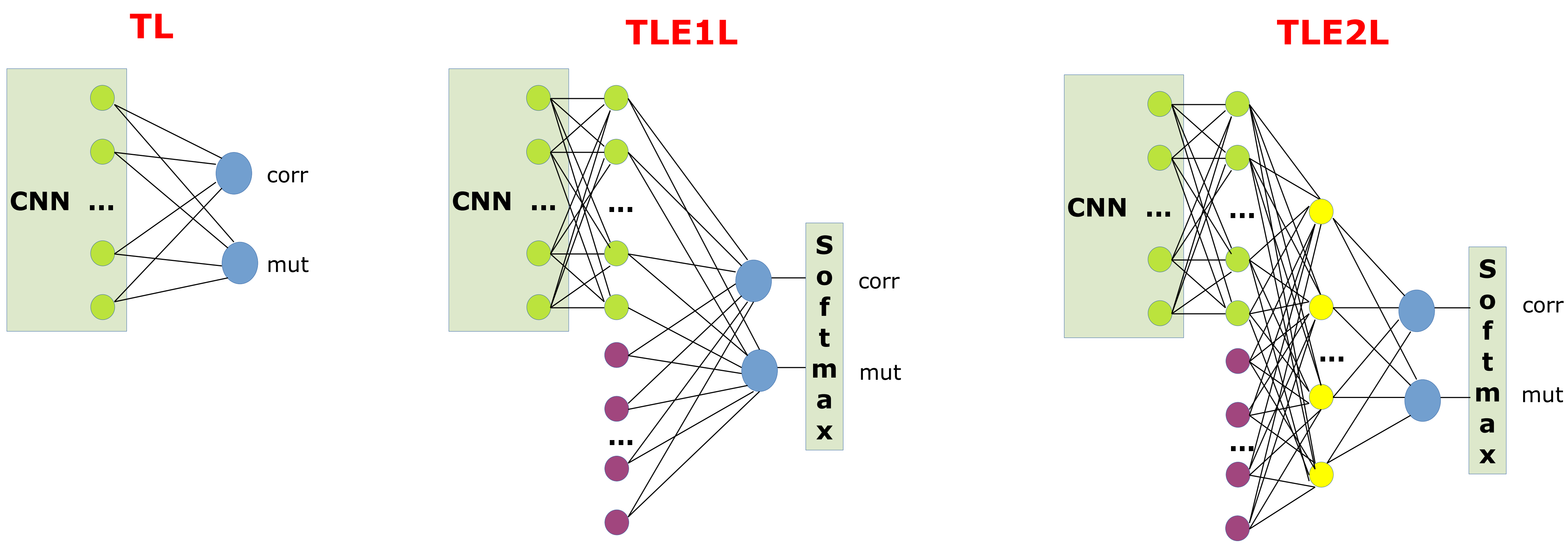
CNN	#Layers	#TL	#TLE1L	#TLE2L	#In Feat
ResNet-18 [20]	18	11.17M	11.44M	11.83M	512
ResNet-34 [20]	34	21.28M	21.55M	21.94M	512
ResNeXt-50-32x4d [62]	50	22.98M	27.18M	27.96M	2,048
Wide ResNet-50-2 [64]	50	66.83M	71.03M	71.81M	2,048
Inception v3 [52]	48	21.78M	25.98M	26.76M	2,048
ResNet-152 [20]	152	58.14M	62.34M	63.12M	2,048
DenseNet-161 [23]	161	26.47M	31.35M	32.17M	2,208

CNNs

Number of millions (M) of trainable parameters.

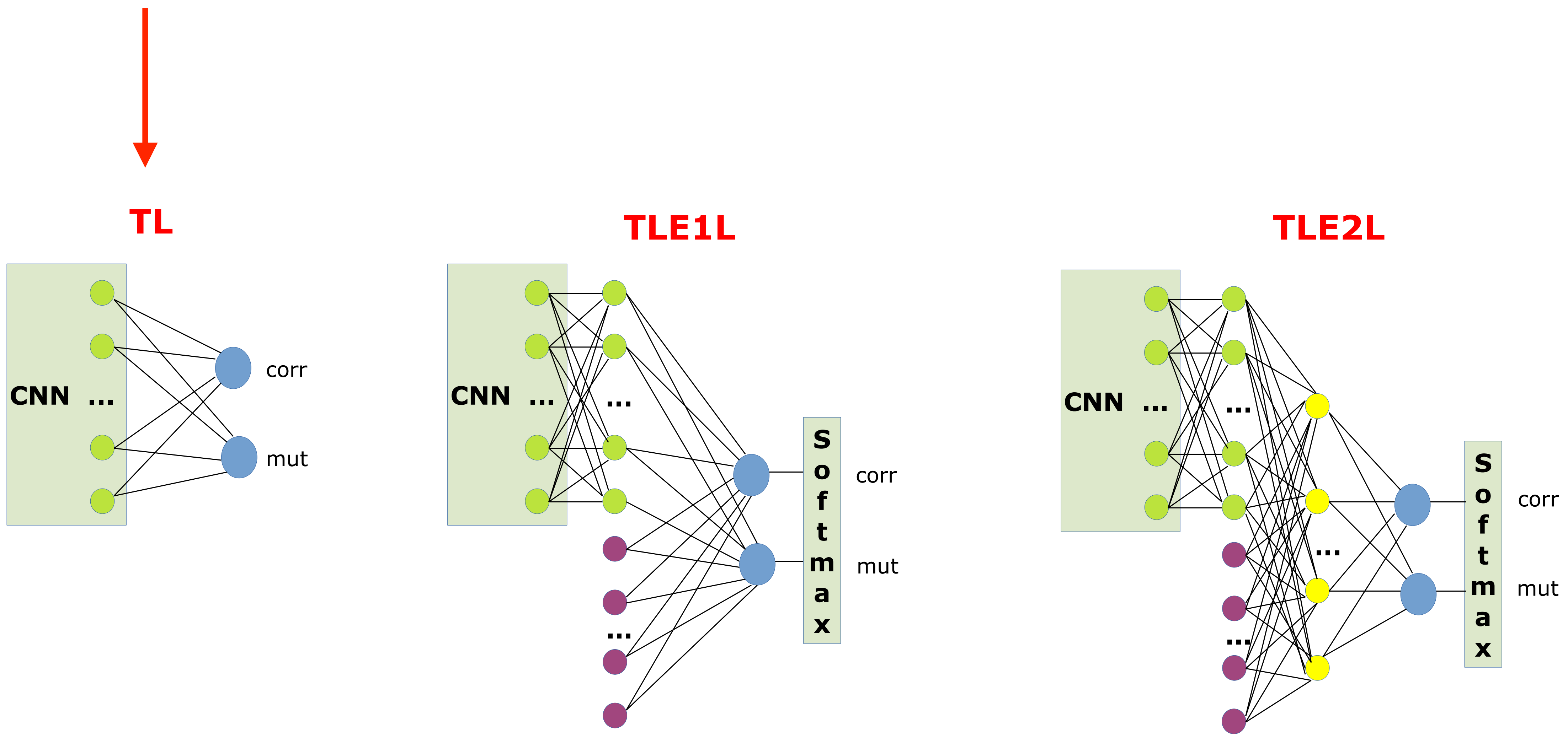
CNN	#Layers	#TL	#TLE1L	#TLE2L	#In Feat
ResNet-18 [20]	18	11.17M	11.44M	11.83M	512
ResNet-34 [20]	34	21.28M	21.55M	21.94M	512
ResNeXt-50-32x4d [62]	50	22.98M	27.18M	27.96M	2,048
Wide ResNet-50-2 [64]	50	66.83M	71.03M	71.81M	2,048
Inception v3 [52]	48	21.78M	25.98M	26.76M	2,048
ResNet-152 [20]	152	58.14M	62.34M	63.12M	2,048
DenseNet-161 [23]	161	26.47M	31.35M	32.17M	2,208

Architecture Configurations



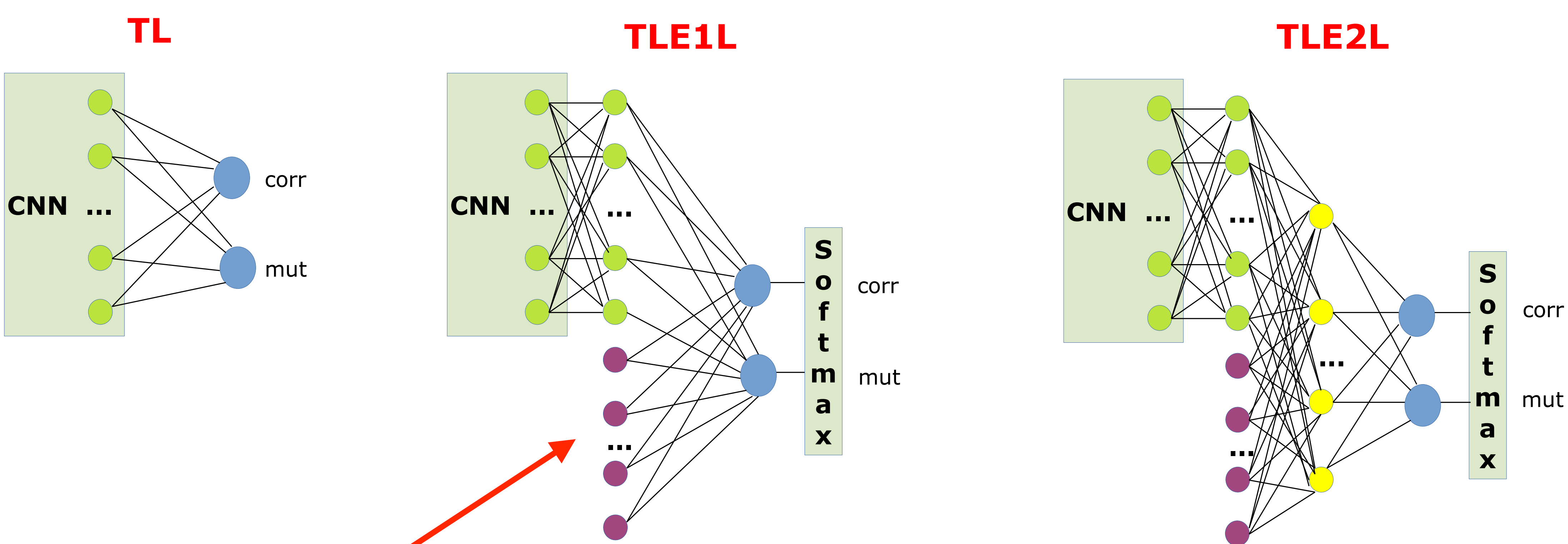
Architecture Configurations

Pure Transfer Learning (TL): as-is configuration.



Architecture Configurations

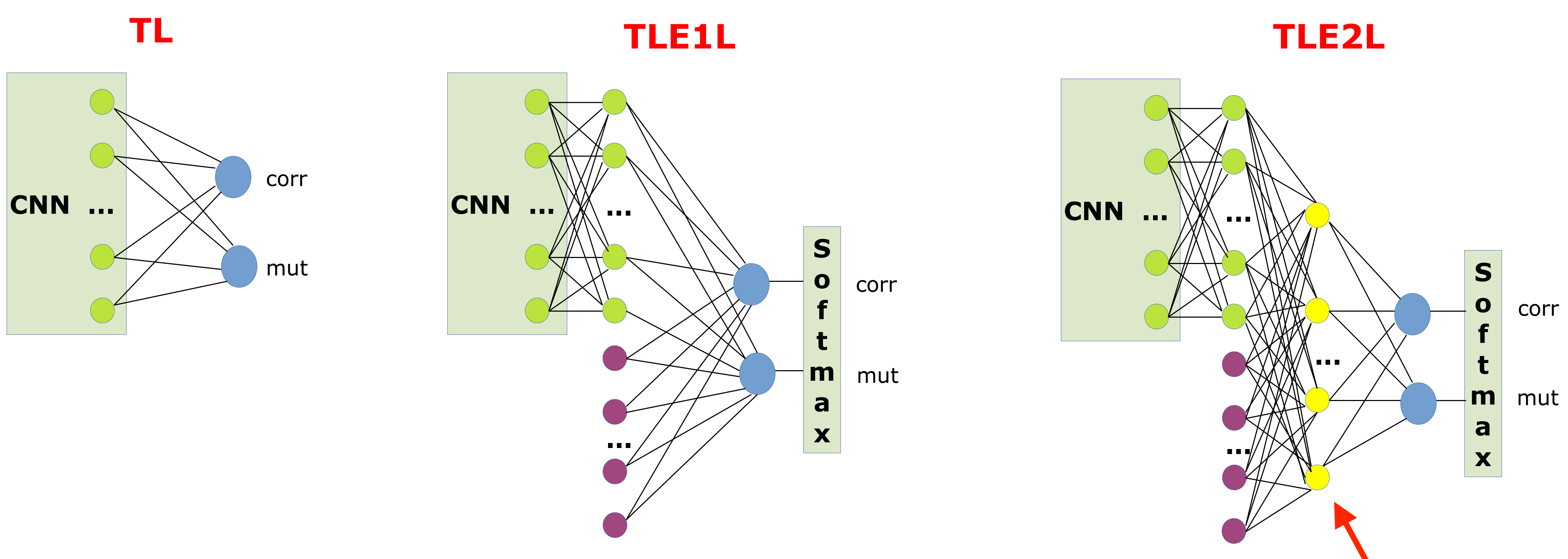
TLE1L: one extra layer.



Feature detector and description algorithm Oriented FAST and Rotated BRIEF (ORB): 1,024 elements.

Architecture Configurations

TLE2L: TLE1L + another extra layer.



256 neurons.

Results and Discussion

CNN	Dataset Profile					
	TD			SS		
	TL	TLE1L	TLE2L	TL	TLE1L	TLE2L
ResNet-18	**0.6125**	**0.6438**	0.625	0.73125	0.74375	0.7625
ResNet-34	0.5188	0.6188	*0.6313*	*0.75*	*0.75*	*0.775*
ResNeXt-50-32x4d	*0.5813*	*0.625*	0.6125	0.6875	*0.75*	0.7625
Wide ResNet-50-2	0.55	0.5625	0.5875	0.675	**0.75625**	0.71875
Inception v3	0.4438	0.6063	0.5875	**0.7875**	**0.75625**	0.7
ResNet-152	*0.5813*	0.55	0.575	*0.75*	0.725	0.7625
DenseNet-161	*0.5813*	0.5438	**0.6375**	0.71875	0.7375	**0.8**

Results and Discussion

Within TD with TL.

CNN	Dataset Profile					
	TD			SS		
	TL	TLE1L	TLE2L	TL	TLE1L	TLE2L
ResNet-18	**0.6125**	**0.6438**	0.625	0.73125	0.74375	0.7625
ResNet-34	0.5188	0.6188	*0.6313*	*0.75*	*0.75*	*0.775*
ResNeXt-50-32x4d	*0.5813*	*0.625*	0.6125	0.6875	*0.75*	0.7625
Wide ResNet-50-2	0.55	0.5625	0.5875	0.675	**0.75625**	0.71875
Inception v3	0.4438	0.6063	0.5875	**0.7875**	**0.75625**	0.7
ResNet-152	*0.5813*	0.55	0.575	*0.75*	0.725	0.7625
DenseNet-161	*0.5813*	0.5438	**0.6375**	0.71875	0.7375	**0.8**

Results and Discussion

Within TD with all architecture configurations.

CNN	Dataset Profile					
	TD			SS		
	TL	TLE1L	TLE2L	TL	TLE1L	TLE2L
ResNet-18	**0.6125**	**0.6438**	0.625	0.73125	0.74375	0.7625
ResNet-34	0.5188	0.6188	*0.6313*	*0.75*	*0.75*	*0.775*
ResNeXt-50-32x4d	*0.5813*	*0.625*	0.6125	0.6875	*0.75*	0.7625
Wide ResNet-50-2	0.55	0.5625	0.5875	0.675	**0.75625**	0.71875
Inception v3	0.4438	0.6063	0.5875	**0.7875**	**0.75625**	0.7
ResNet-152	*0.5813*	0.55	0.575	*0.75*	0.725	0.7625
DenseNet-161	*0.5813*	0.5438	**0.6375**	0.71875	0.7375	**0.8**

RQ_1: Weighted Ranking

- 1. **DenseNet-161.**
- 2. **ResNet-18 and Inception v3 (tie).**
- 4. ResNet-34.
- 5. ResNeXt-50-32x4d.
- 6. Wide ResNet-50-2.
- 7. ResNet-152.

Answering RQ_1

- Does a deeper CNN (more layers) always have better performance compared to a shallower (less layers) one?

- R: **A deeper CNN does not necessarily have better performance than a shallower one. When reusing pretrained models to address a new problem (as the test oracle task we did here), it is recommended to eventually start with shallower networks, which usually have smaller number of trainable parameters and usually demand less powerful computational infrastructure.**

Possible Recommendation

- **DenseNet-161** was also the best here (classification, Cerrado images, 11 DNNs):
  - M. S. Miranda, L. F. A. Silva, S. F. dos Santos, V. A. Santiago Júnior, T. S. Körting, and J. Almeida. **A High-Spatial Resolution Dataset and Few-shot Deep Learning Benchmark for Image Classification**. In: The 35th Conference on Graphics, Patterns and Images (SIBGRAPI 2022), 2022, Natal, RN, Brazil. Accepted for publication.

Source: https://github.com/ai4luc/CerraData-code-data

RQ_2: Transfer Learning

TL X max(TLE1L, TLE2L): Only in two out of 14 situations there was a decrease in the accuracy.

CNN	Dataset Profile					
	TD			SS		
	TL	TLE1L	TLE2L	TL	TLE1L	TLE2L
ResNet-18	**0.6125**	**0.6438**	0.625	0.73125	0.74375	0.7625
ResNet-34	0.5188	0.6188	*0.6313*	*0.75*	*0.75*	*0.775*
ResNeXt-50-32x4d	*0.5813*	*0.625*	0.6125	0.6875	*0.75*	0.7625
Wide ResNet-50-2	0.55	0.5625	0.5875	0.675	**0.75625**	0.71875
Inception v3	0.4438	0.6063	0.5875	**0.7875**	**0.75625**	0.7
ResNet-152	*0.5813*	0.55	0.575	*0.75*	0.725	0.7625
DenseNet-161	*0.5813*	0.5438	**0.6375**	0.71875	0.7375	**0.8**

RQ_2: Transfer Learning

TD, TLE1L, Inception v3: increase of 36.62% in the accuracy.

CNN	Dataset Profile					
	TD			SS		
	TL	TLE1L	TLE2L	TL	TLE1L	TLE2L
ResNet-18	**0.6125**	**0.6438**	0.625	0.73125	0.74375	0.7625
ResNet-34	0.5188	0.6188	*0.6313*	*0.75*	*0.75*	*0.775*
ResNeXt-50-32x4d	*0.5813*	*0.625*	0.6125	0.6875	*0.75*	0.7625
Wide ResNet-50-2	0.55	0.5625	0.5875	0.675	**0.75625**	0.71875
Inception v3	0.4438	0.6063	0.5875	**0.7875**	**0.75625**	0.7
ResNet-152	*0.5813*	0.55	0.575	*0.75*	0.725	0.7625
DenseNet-161	*0.5813*	0.5438	**0.6375**	0.71875	0.7375	**0.8**

Answering RQ_2

- If we do not change the architecture of a predefined model/network, is pure transfer learning able to get the same or better performances compared to extended architectures of the model?

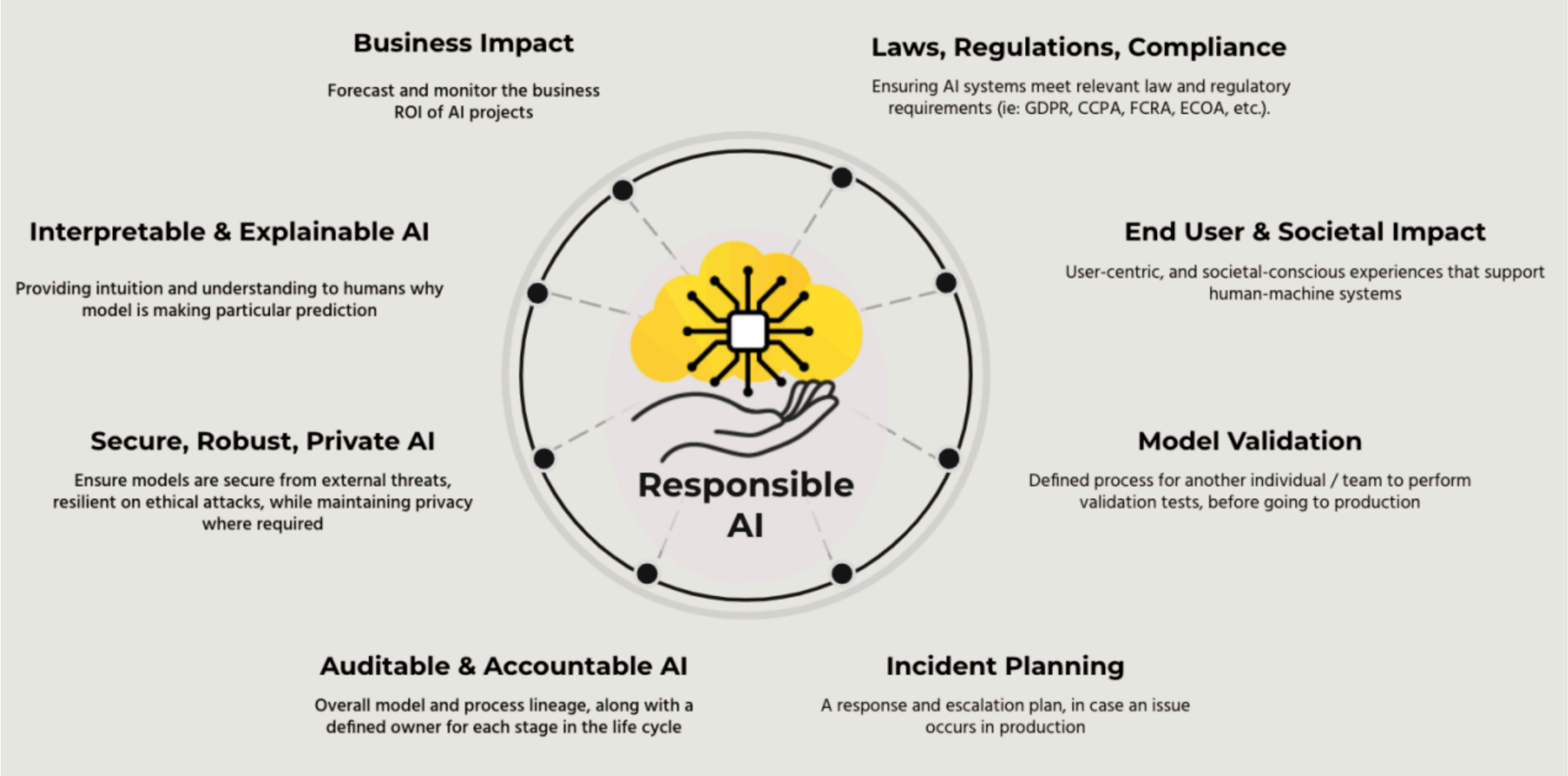
- R: **Pure transfer learning is a valuable technique within DNNs but eventually we have to extend previous model's architectures to get better results. Moreover, the related domain requirement seems to be crucial.**

Richard Feynman

- Nobel Prize in Physics (1965): “What I cannot create, I do not understand”.

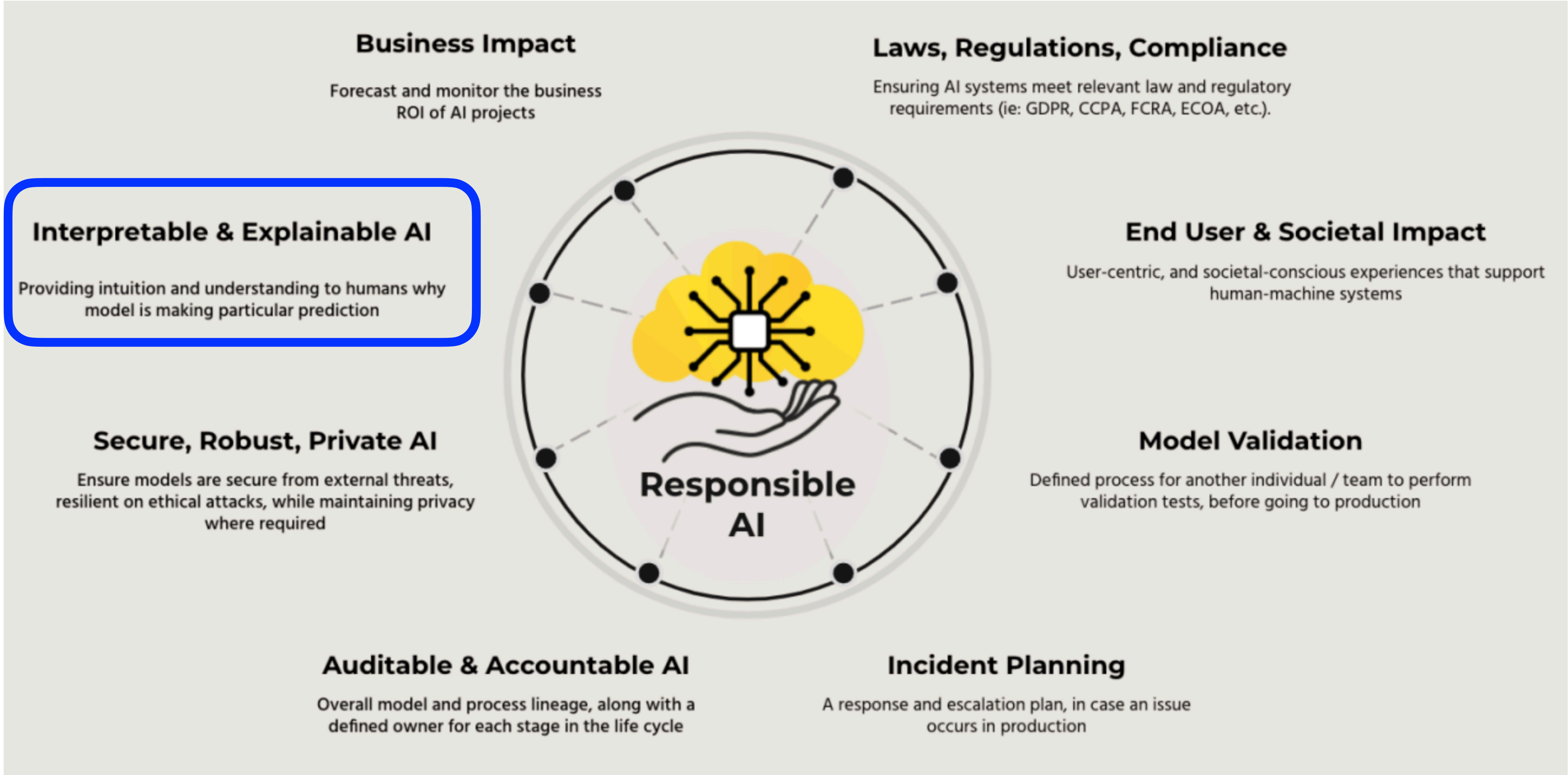
Explainability!

Responsible AI



Source: https://h2o.ai/insights/responsible-ai/

Explainable AI (XAI)



Source: https://h2o.ai/insights/responsible-ai/

XAI: DARPA

DARPA DEFENSE ADVANCED RESEARCH PROJECTS AGENCY

EXPLORE BY TAG

ABOUT US / OUR RESEARCH / NEWS / EVENTS / WORK WITH US /

› Defense Advanced Research Projects Agency › Our Research › Explainable Artificial Intelligence

Explainable Artificial Intelligence (XAI)

Dr. Matt Turek

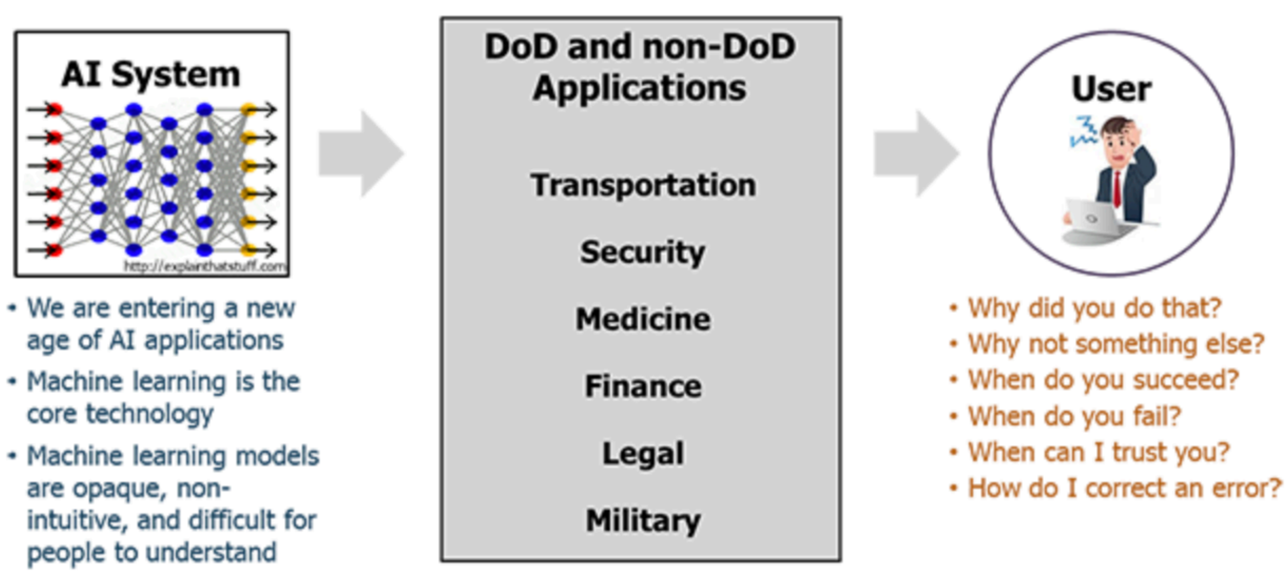


Figure 1. The Need for Explainable AI

RESOURCES

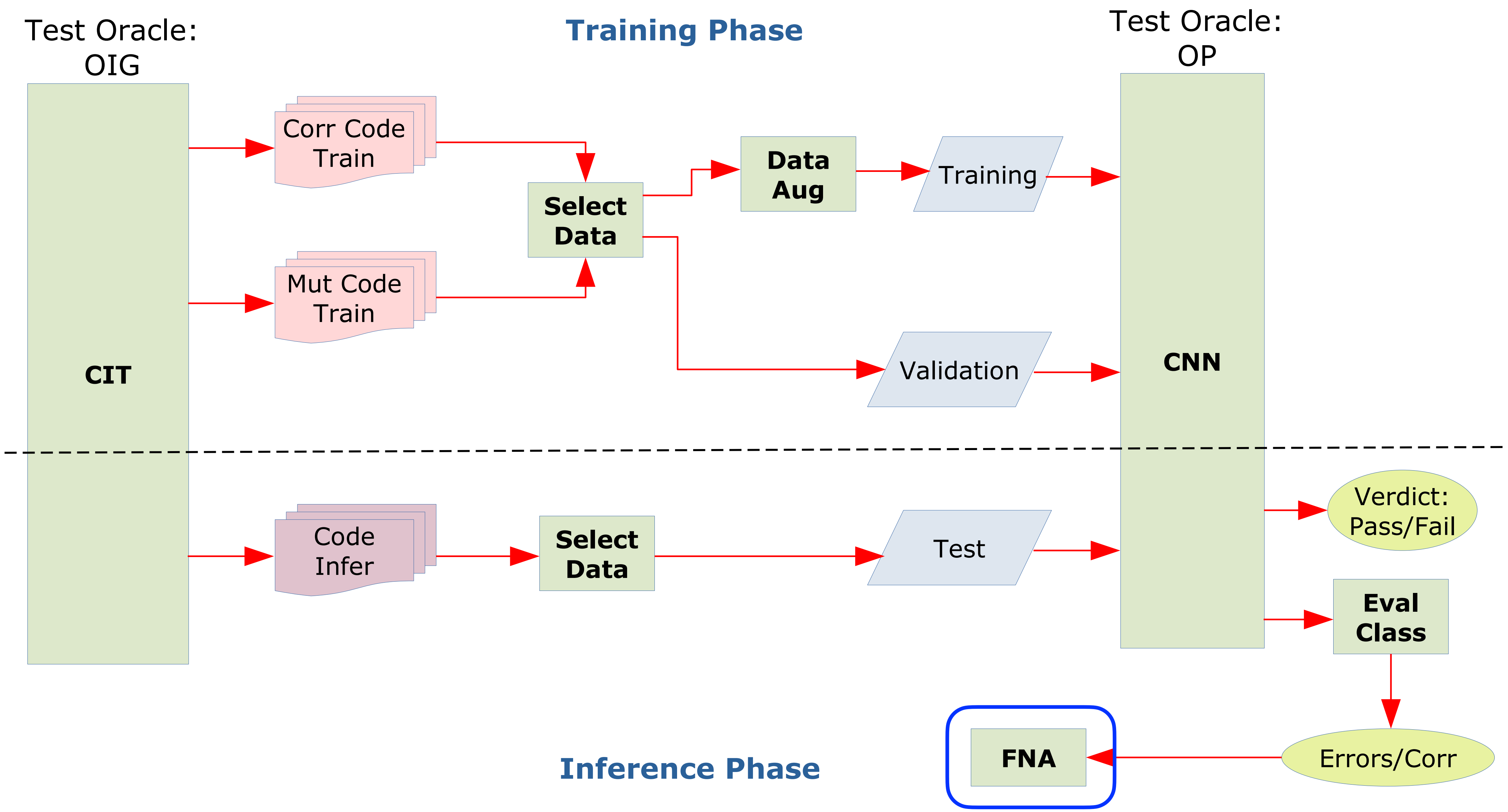
DARPA-BAA-16-53

DARPA-BAA-16-53: Proposers Day Slides

XAI Program Portfolio

Source: https://www.darpa.mil/program/explainable-artificial-intelligence

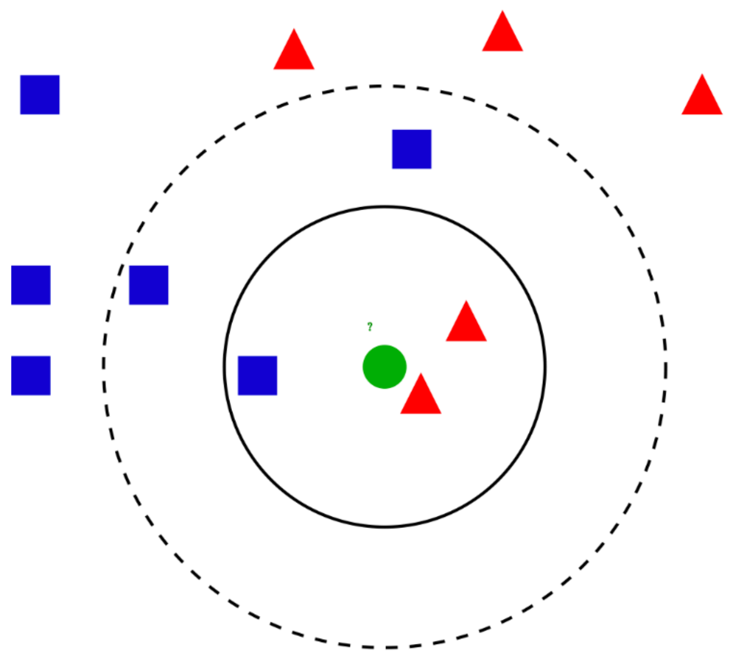
TOrC: Evaluate Classification



The FNA Technique

- FNA: straightforward and black-box approach relying only on the images of the training and test sets.

- FNA: based on the K-nearest neighbours (KNN) ML algorithm.



Source: https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/

The FNA Technique

Algorithm 1 The FNA technique

Input: T, S

Output: P

1: $F = T \cup S$
2: $t = |T|$
3: $s = |S|$
4: $n = \lfloor (t + s)/s \rfloor$
5: $K = findNearestNeighbours(n, F)$
6: $X = countNumberNeighbours(K, S)$
7: $X = X/n$
8: $P = findMaxProportion(X, S)$
9: **return** P

As for FNA, we define six classes:

tr_cor;

tr_mut;

mi_cor;

mi_mut;

co_cor;

co_mut.

The FNA Technique

Algorithm 1 The FNA technique

Input: T, S

Output: P

1: $F = T \cup S$

2: $t = |T|$

3: $s = |S|$

4: $\boxed{n = \lfloor (t + s)/s \rfloor}$

5: $K = findNearestNeighbours(n, F)$

6: $X = countNumberNeighbours(K, S)$

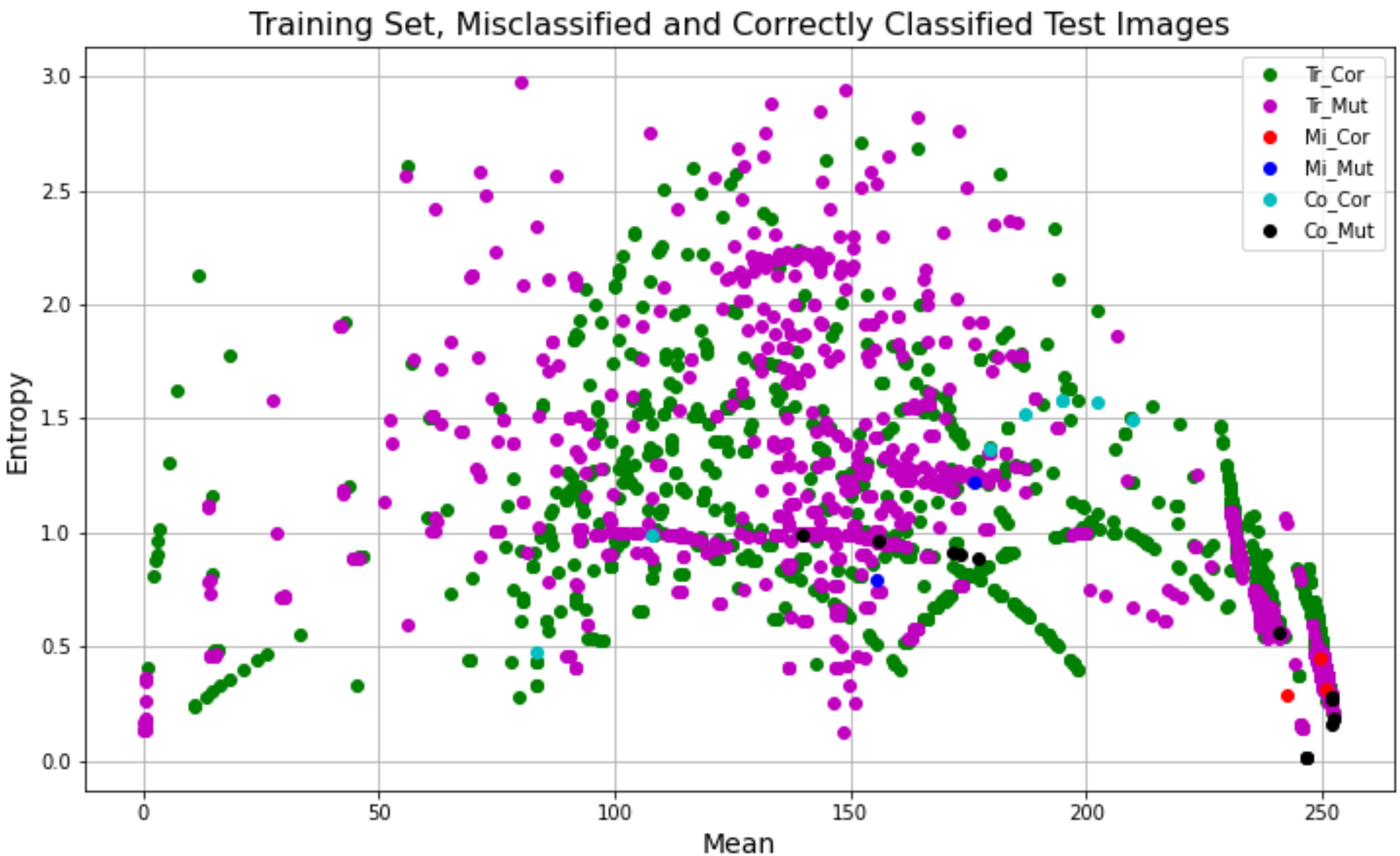
7: $X = X/n$

8: $P = findMaxProportion(X, S)$

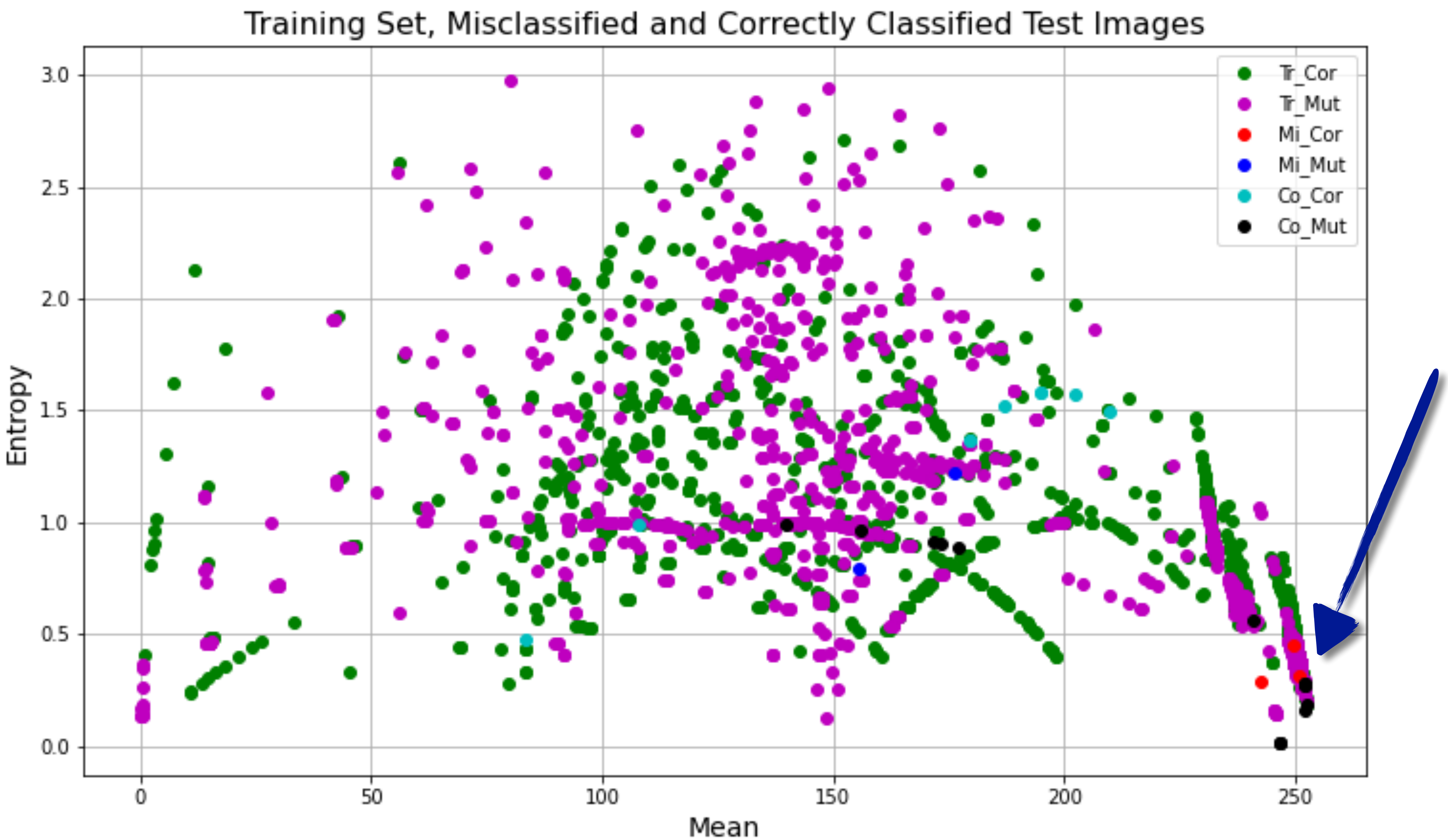
9: **return** P

Define the number of nearest neighbours, n, for each image i_s of the test set, where each image i_s is viewed as a centroid of a cluster.

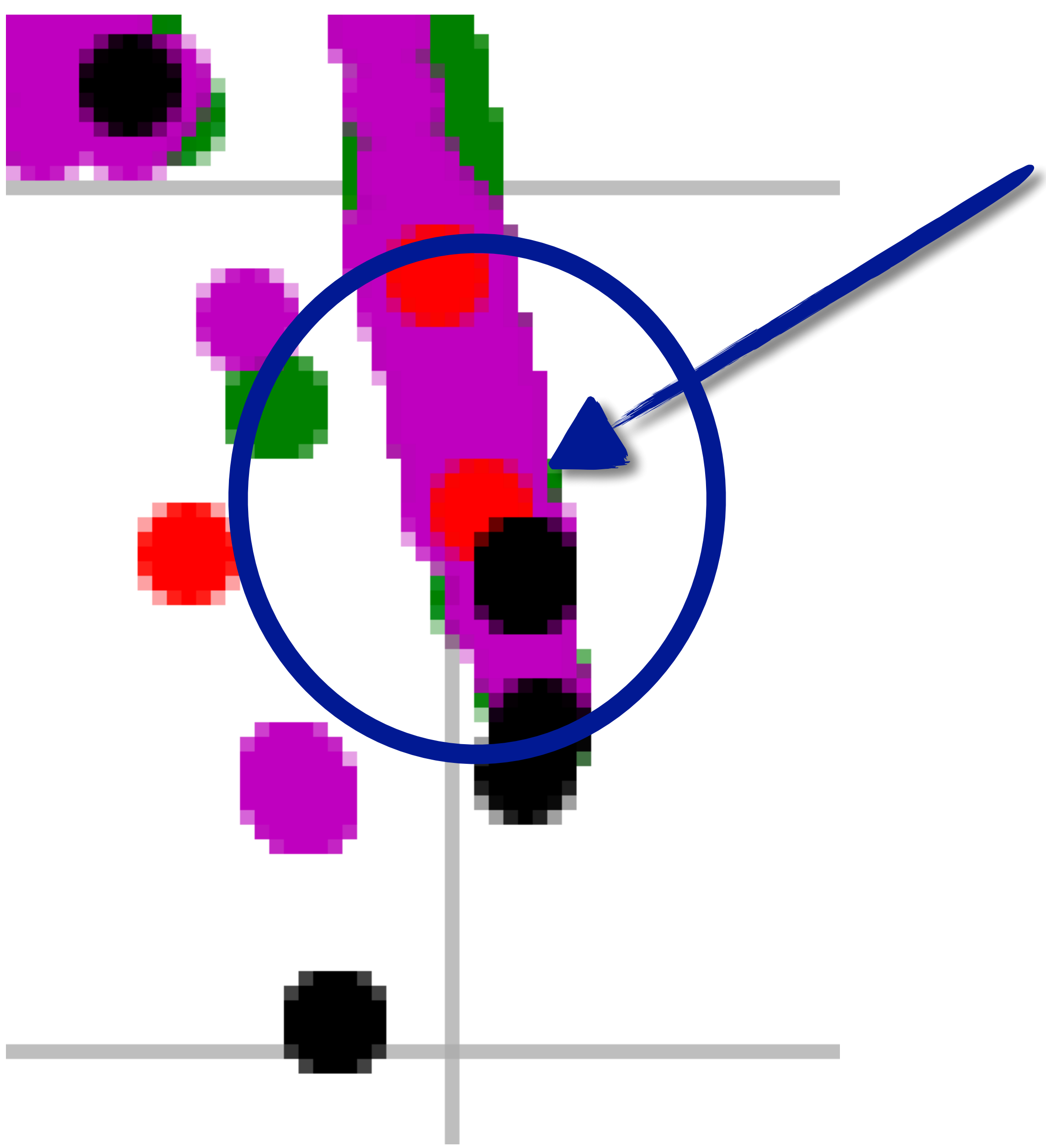
The FNA Technique



The FNA Technique



The FNA Technique



If an image of the correct class of the test set was misclassified (*mi_cor*), we would expect that the corresponding element in *P* is *tr_mut*.

FNA: Evaluation

- Three best models: DenseNet-161, ResNet-18, and Inception v3.

- Entire training set of both profiles (TD and SS) and the 29 corner case images of the test set.
  - Five of these test images: misclassified by all 18 combinations of model, dataset profile, and architecture configuration;
  - Remaining 24 images: correctly classified by all 18 combinations.

FNA: Evaluation

- Image features: mean, Shannon entropy, contrast, dissimilarity, homogeneity, correlation, and angular second-moment.

- Number of neighbours, n, in TD is 93 and 99 in SS.

FNA: Evaluation

- For both profiles, TD and SS, we got the same result. In **only one** (same image) out of the 29 corner case images FNA failed.

- FNA's accuracy: $28/29 = 0.9655$.

To sum up

- Fields, techniques related to this research:
  - Software testing (test oracle, CIT, mutation analysis);
  - Deep learning;
  - Deep convolutional neural networks (CNNs);
  - Transfer learning;
  - Explainable artificial intelligence;
  - Data-centric artificial intelligence;

To sum up

- Fields, techniques related to this research (cont):
  - Data augmentation;
  - Image similarity metrics (structural similarity, Fréchet Inception Distance (FID));
  - Image features;
  - Oriented FAST and Rotated BRIEF (ORB) algorithm;
  - K-nearest neighbours (KNN);
  - Apriori algorithm.

Article

A Method and Experiment to evaluate Deep Neural Networks as Test Oracles for Scientific Software

Publisher: IEEE Cite This PDF

Valdivino Alexandre de Santiago Júnior **All Authors**

Abstract

Document Sections

Abstract:
Testing scientific software is challenging because usually such type of systems have non-deterministic behaviours and, in addition, they generate non-trivial outputs such as images. Artificial intelligence (AI) is now a reality which is also helping in the development of the software testing activity. In this article, we evaluate seven deep neural networks (DNNs), precisely deep convolutional neural networks (CNNs) with up to 161layers, playing the role of test oracle procedures for testing scientific models. Firstly, we propose a method, TOrC, which starts by generating training, validation, and test image datasets via combinatorial interaction testing applied to the original codes and second-order mutants. Within TOrC we also have classical steps such as transfer learning, a technique recommended for DNNs. Then, we verified the performance of the oracles (CNNs). The main conclusions of this research are: i) not necessarily a greater number of layers means that a CNN will present better performance; ii) transfer learning is a valuable technique but eventually we may need extended solutions to get better performances; iii) data-centric AI is an interesting path to follow; and iv) there is not a clear correlation between the software bugs, in the scientific models, and the errors (image misclassifications) presented by the CNNs. CCS CONCEPTS • Software and its engineering → Software testing and debugging;. Computing methodologies → Neural networks; Supervised learning by classification; Computer vision.

Published in: 2022 IEEE/ACM International Conference on Automation of Software Test (AST)

Source: https://ieeexplore.ieee.org/document/9796455

Thank You!

E-mail: valdivino.santiago@inpe.br

Web: http://www.lac.inpe.br/~valdivino/

GitHub: https://github.com/vsantjr

What to do?

- Detailed analysis of the features/characteristics of the images in the sets (training, validation, test).
- Generate more images (data augmentation; GANs).
- Trying different splittings (training, validation, test).
- Tuning of hyper-parameters.
- "Mosaic" data augmentation. Center cropping (224x224) makes more difficult the job of the learner.
- Selection of another model rather than CNN.

References

- [Mahajan and Halfond 2015/ICST]. S. Mahajan and W. G. J. Halfond. 2015. Detection and Localization of HTML Presentation Failures Using Computer Vision-Based Techniques. In 2015 IEEE 8th International Conference on Software Testing, Verification and Validation (ICST). 1–10. https://doi.org/10.1109/ICST.2015.7102586
- [Kiraç et al. 2018/JSS]. M. F. Kiraç, B. Aktemur, and H. Sözer. 2018. VISOR: A fast image processing pipeline with scaling and translation invariance for test oracle automation of visual output systems. Journal of Systems and Software 136 (2018), 266 – 277. https://doi.org/10.1016/j.jss.2017.06.023

References

- [Frounchi et al. 2011/IST]. K. Frounchi, L. C. Briand, L. Grady, Y. Labiche, and R. Subramanyan. 2011. Automating image segmentation verification and validation by learning test oracles. Information and Software Technology 53, 12 (2011), 1337–1348. https://doi.org/10.1016/j.infsof.2011.06.009
- [Santiago Júnior et al. 2018/SAST]. V. A. Santiago Júnior, L. A. R. Silva, and P. R. Andrade Neto. 2018. Testing Environmental Models Supported by Machine Learning. In Proceedings of the III Brazilian Symposium on Systematic and Automated Software Testing (SAO CARLOS, Brazil) (SAST '18). Association for Computing Machinery, New York, NY, USA, 3–12. https://doi.org/10.1145/3266003.3266004

References

- [Liu et al. 2020/ASE]. Z. Liu, C. Chen, J. Wang, Y. Huang, J. Hu, and Q. Wang. 2020. Owl Eyes: Spotting UI Display Issues via Visual Understanding. Association for Computing Machinery, New York, NY, USA, 398–409. https://doi.org/10.1145/3324884.3416547
- [Pan et al. 2020/ISSTA]. M. Pan, A. Huang, G. Wang, T. Zhang, and X. Li. 2020. Reinforcement Learning Based Curiosity-Driven Testing of Android Applications. In Proceedings of the 29th ACM SIGSOFT International Symposium on Software Testing and Analysis (Virtual Event, USA) (ISSTA 2020). Association for Computing Machinery, New York, NY, USA, 153–164. https://doi.org/10.1145/3395363.3397354

References

- [Miranda et al. 2021/ICCSA]. M. S. Miranda, V. A. Santiago Júnior, T. S. Körting, R. Leonardi, and M. L. Freitas Júnior. Deep Convolutional Neural Network for classifying Satellite Images with Heterogeneous Spatial Resolutions. In: Gervasi, O. et al. (eds), Computational Science and Its Applications - ICCSA 2021. Lecture Notes in Computer Science, vol 12955, p. 519-530, Springer, Cham
- [Balera and Santiago Júnior 2017/JSERD]. J. M. Balera and V. A. Santiago Júnior. 2017. An algorithm for combinatorial interaction testing: definitions and rigorous evaluations. Journal of Software Engineering Research and Development 5, 1 (28 Dec 2017), 10. https://doi.org/10.1186/s40411-017-0043-z