



A Principled Approach to Failure Analysis and Model Repairment: Demonstration in Medical Imaging

Thomas Henn¹(✉), Yasukazu Sakamoto², Clément Jacquet¹, Shunsuke Yoshizawa², Masamichi Andou², Stephen Tchen¹, Ryosuke Saga¹, Hiroyuki Ishihara², Katsuhiko Shimizu², Yingzhen Li³, and Ryutaro Tanno⁴

¹ Rokken Inc., Osaka, Japan

`thomas.henn@lab6.co.jp`

² Corporate R&D Center, Terumo Corporation, Tokyo, Japan

`yasukazu_sakamoto@terumo.co.jp`

³ Department of Computing, Imperial College London, London, UK

⁴ Department of Computing, University College London, London, UK

Abstract. Machine learning models commonly exhibit unexpected failures post-deployment due to either data shifts or uncommon situations in the training environment. Domain experts typically go through the tedious process of inspecting the failure cases manually, identifying failure modes and then attempting to fix the model. In this work, we aim to standardise and bring principles to this process through answering two critical questions: (i) how do we know that we have identified meaningful and distinct failure types?; (ii) how can we validate that a model has, indeed, been repaired? We suggest that the quality of the identified failure types can be validated through measuring the intra- and inter-type generalisation after fine-tuning and introduce metrics to compare different subtyping methods. Furthermore, we argue that a model can be considered repaired if it achieves high accuracy on the failure types while retaining performance on the previously correct data. We combine these two ideas into a principled framework for evaluating the quality of both the identified failure subtypes and model repairment. We evaluate its utility on a classification and an object detection tasks. Our code is available at <https://github.com/Rokken-lab6/Failure-Analysis-and-Model-Repairment>.

Keywords: Failure analysis · Model repairment · Deep learning

1 Introduction

It is common for new failures to be discovered once a model has been deployed in the “wild”. While recent lines of research in medical imaging have shown promising results in designing robust machine learning (ML) models [1–5], it may not be realistic to achieve perfect generalisation to every relevant environment.

© Springer Nature Switzerland AG 2021

M. de Bruijne et al. (Eds.): MICCAI 2021, LNCS 12903, pp. 509–518, 2021.

https://doi.org/10.1007/978-3-030-87199-4_48

Consequently, recently published guidelines for the reliable application of ML systems in healthcare [6, 7] and recent work from Luke *et al.* [8] stress the importance of analyzing and reporting clinically relevant failure cases. However, there is a lack of standardised protocols to identify, validate and analyze those failure types. Typically, domain experts manually inspect the failure cases and make sense of them by identifying a set of failure modes. But this approach can be both expensive and biased by the human expertise. For example, a critical yet rare subgroup could be missed with such an approach and go unreported [8]. A notable recent work [9] recognises this issue and makes a first step towards data-driven approaches to failure subtyping through clustering of the learned features based on whether its presence or absence is predictive of poor performance. However to date, little attention has been gathered around the evaluation metrics of the identified failure types, hampering the development of new methods in this direction. Furthermore, even if a set of meaningful failure types could be identified, methods for *fixing* them and evaluating its success remain undeveloped.

In an attempt to bring principles to the process of failure analysis and model repairment, we introduce a framework for not only deriving subtypes of failure cases and measuring their quality, but also repairing the models and verifying their generalisation. We put forward a set of desirable properties that a meaningful set of failure subtypes should meet and design surrogate metrics. Moreover, we propose a data-driven method for identifying failure types based on clustering in feature or gradient spaces. This method was able not only to identify failure types with the highest quality according to our metrics but also to identify clinically important failures like undetected catheters close to the ultrasound probe in intracardiac echocardiography. Finally, we argue that model repairment should not only aim to fix each failure type in a generalizable manner but also to ensure the performance on previously successful cases is retained.

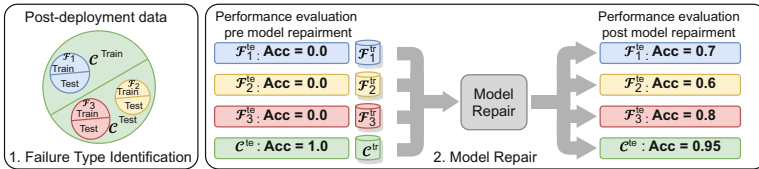


Fig. 1. An overview of the proposed failure analysis and model repairment framework which proceeds in two phases. Firstly, based on the evaluation of the model \mathcal{M} , the set of “failure cases” \mathcal{F} (e.g., misclassified cases, examples with high errors, etc.) are clustered to identify failures types $\{\mathcal{F}_i\}_i$ such that $\mathcal{F} = \cup_i \mathcal{F}_i$. Secondly, the failure types $\{\mathcal{F}_i\}_i$ are “fixed” by a *repairment* algorithm (e.g., fine-tuning, continual learning methods, etc.) based on a “training split” of the failure sets $\{\mathcal{F}_i^{tr}\}$, and the correct set \mathcal{C}^{tr} . Finally, the success of the repairment is evaluated by measuring generalisation on the test sets of target failure types $\{\mathcal{F}_i^{te}\}$ and the correct cases \mathcal{C}^{te} .

2 Methods

We elaborate below the two phases (see Fig. 1) of our approach to model repair.

2.1 Phase I: Identification of Failure Types

It is not evident how to optimally separate failure cases into a set of distinct failure types. Domain experts could split the failure cases according to the visual appearance or consider the importance of different failures from a clinical perspective, for example according to stages or clinical signs of a disease. However, we suggest clinical relevance doesn't necessarily reflect the way the model distinctly fails on each failure type. Failure types should be specific not only to the model but the repairment methods that can be used to fix them. Moreover, objective metrics to quantify the quality of failure types for the purpose of model repairment are still lacking.

Two Desiderata of Failure Types: We postulate that a set of failure types should satisfy two desirable properties: *Independence* and *Learnability*. In addition, we propose two novel surrogate metrics to assess those properties in practice. They are measured by fine-tuning the model \mathcal{M} on a subset \mathcal{F}_i^{tr} of the given failure type \mathcal{F}_i under a *Compatibility constraint*, and then calculating the performance on each \mathcal{F}_j^{te} . We assume a sufficiently large set of failures with no label noise or corrupted images.

- (i) *Independence:* The subtypes should be as independent as possible from each other in the way the model fails. In other words, they can be fixed by comparatively distinct alterations to the decision boundary. If two types are independent, a model fine-tuned on one type should not be useful to the other. In practice, we suggest a continuous measure of *Independence* by calculating the average difference in the test performance between each failure type and the rest: $I(\mathcal{F}_i) := \text{mean}_{j \neq i} (m(\mathcal{M}_i, \mathcal{F}_i^{te}) - m(\mathcal{M}_i, \mathcal{F}_j^{te}))$ where $m(\cdot)$ is a performance metric (e.g., accuracy).
- (ii) *Learnability:* Each subtype should be homogeneous and consists of examples for which the model has failed in a similar way. In other words, a such failure type contains failure cases that can be fixed via a similar modification to the model's decision boundary. If a subtype is heterogeneous, fixing it would require more modifications to the model and would be more challenging. We thus posit that a more homogeneous failure type would be easier to learn, and measure *Learnability* as the generalisation of the fine-tuned model \mathcal{M}_i on the chosen failure type $L(\mathcal{F}_i) := m(\mathcal{M}_i, \mathcal{F}_i^{te})$.

Compatibility Constraint: We argue in addition that the above surrogate metrics should be measured with the constraint of maintaining performance on correct data \mathcal{C} . This is necessary to avoid learning pathological discriminative rules just to solve a specific failure type. For example, given a failure type only containing

images of a single class, the model could simply learn to ignore the input images and predict the same class everywhere, in which case the failure is fixed in a meaningless way. *Compatibility* ensures “locality” of the failure types by ensuring the required changes in the discriminative rules do not considerably influence the previous cases. *Compatibility* is achieved by fine-tuning on both a failure type \mathcal{F}_i^{tr} and previously correct cases \mathcal{C}^{tr} in equal proportions and early stopping when \mathcal{C}^{tr} validation performance drops below 0.9.

Automatic Identification of Failure Types: Manual analysis can often be time-consuming but also suboptimal, potentially overlooking meaningful failure modes. Therefore, we wish to automatically uncover a set of failure types with good *Independence* and *Learnability* scores. We thus also explore two methods as specific instantiations of our framework; particularly, we experiment with clustering the failure cases \mathcal{F} in the feature space (*Feature clustering*) and gradient space of a differentiable model \mathcal{M} (*Gradient clustering*). The gradients of the loss with respect to the parameters are used (for object detection: the loss specific to the object). We expect that similar data will be close in the feature space and the failures whose correction requires similar changes to the model parameters will be close in the gradient space. Furthermore, features are averaged over spatial dimensions and both features and gradients are reduced by Gaussian random projection followed by UMAP [10] to 10 dimensions. Finally, the data is clustered through k-means with the highest Silhouette score between 3 and 10 clusters.

2.2 Phase II: Repairment of Failure Types

Once a set of target failure types has been identified, the model needs to be repaired. We argue that a successful repairment leads to a model that generalizes on unseen cases of the target failure types while maintaining performance on the cases \mathcal{C} where the model previously performed well. First, a target set of failure types $\mathcal{F}^T \subset \{\mathcal{F}_i\}_i$ is selected by the end-users, and then each type is split into two sets, where one is used to first repair the model (e.g. \mathcal{F}_i^{tr}) and then the other is used to evaluate generalization (e.g. \mathcal{F}_i^{te}). For repairment procedures, we experiment with fine-tuning on the failure types (with or without the correct cases) and *elastic weight consolidation* (EWC), a popular continual learning approach [11]. We note, however, that more recent model adaptation approaches are also applicable in our framework such as more recent variants of continual learning [12, 13], meta-learning [14] and domain adaptation [15].

3 Datasets and Implementation Details

We evaluate the efficacy of the proposed model repairment framework on two medical imaging datasets. The details of the respective datasets along with the specification of models/optimisation are provided below.

Binary PathMNIST (BP-MNIST) is a publicly available classification dataset consisting of colorectal cancer histology slides patches [16, 17] derived originally from the NCT-CRC-HE-100K [17] dataset and resized to $3 \times 28 \times 28$ as a part of the MedMNIST benchmark [16]. We simplify the original 9-way classification task into a binary task of discriminating benign and malignant classes (*cancer-associated stroma* and *colorectal adenocarcinoma epithelium*) and use the original classes of granular tissue types as metadata to interpret the discovered failure types. Moreover, 40% of the dataset was put into the test set to increase the sample size for the evaluation of both subtyping and model repairment. Finally, the model was trained with Adam with a learning rate of 10^{-4} in combination with early stopping on the validation accuracy. The architecture is a version of VGG [18] with 6 convolutional layers starting at 16 channels and the fully connected layer replaced by a 1×1 convolution to two output channels and a spatial average.

ICE Catheter Detection (ICE-CD) is a private real-world object detection dataset comprised of ultrasound images of intra-cardiac catheters made by a Intracardiac Echocardiography (ICE) device on pigs. Furthermore, for the purpose of evaluating the performance of catheter detection models, each catheter image was classified into different types representing known difficult situations based on catheter appearance or position. In addition, information about the rough anatomical locations of the probe is available as metadata. The architecture is composed of 5 residual blocks of two convolutional layers (starting at 8 channels and doubling up to 128 channels) followed by two 1×1 convolutions branches: a classification and a center position regression branch. This dataset has been acquired in accordance with animal experiment regulations.

4 Experiments and Results

4.1 Comparison of Failure Subtyping Methods

Baselines and Experiments: we aim to quantify the quality of the proposed automatic subtyping methods (see Table 1 and Fig. 2) which we compare against several baselines: *Random* (random clusters), *False positives and negatives (FP/FN)* (two clusters) and *Metadata* (BP-MNIST: the original 9 *tissue types*; ICE-CD: *image types* as identified by the clinicians and in addition *anatomical locations* of the images). For each failure type \mathcal{F}_i , the model was fine-tuned on both \mathcal{F}_i^{tr} and the correct cases \mathcal{C}^{tr} (to satisfy *Compatibility*). Early stopping is performed with the best validation score (accuracy on BP-MNIST and F_1 on ICE-CD) on \mathcal{F}_i^{tr} while maintaining 0.9 validation accuracy on \mathcal{C}^{tr} . Table 1 displays the average metrics for the respective methods while Fig. 2 shows granular results i.e., matrix $m(\mathcal{M}_i, \mathcal{F}_j^{te})$ which denotes the test accuracy on \mathcal{F}_j^{te} of the model fine-tuned on \mathcal{F}_i^{tr} .

Analysis: first of all, *Gradient clustering* reached better scores than any other method with the exception of *Independence* for *FP/FN* clustering on BP-MNIST. However, it had a 18% higher Learnability score and is more informative

with more identified failure types. Remarkably, *Gradient clustering* was better than using *Metadata*, including the ICE-CD metadata made through very time consuming visual inspection. In the case of BP-MNIST, the lack of independence of *Metadata* subtyping was clearly visible in Fig. 2(a). This implies that *Gradient clustering* might be able to identify independent failure types which are not obvious through human eyes but are relevant to the model. On the other hand, *Feature clustering* seemed to achieve lower scores than metadata. Furthermore, *Gradient clustering* might achieve higher *Learnability* and *Independence* due to being more aligned with the repairment method. Finally, *Random* resulted in by far the lowest independence scores as is apparent in Fig. 2(c) where all types had the same score. This shows that the *Independence* metric is effective in detecting when failure types are mixed together. Moreover, *Random* and *FP/FN* clustering showed lower *Learnability* which may be explained by the diversity of tasks to be learned within each cluster. However, *FP/FN* had the highest *Independence* due to matching the two classes.

Table 1. Comparison of methods for failure types identification on both BP-MNIST and ICE-CD. The best and the second best results are shown in red and blue.

Method	BP-MNIST (ACC)		Catheter Detection (ACC)	
	Learnability	Independence	Learnability	Independence
Random	0.42±0.01	0.00±0.02	0.65±0.02	0.01±0.01
False positives and negatives	0.74±0.22	0.74±0.22	0.69±0.14	0.14±0.37
BP-MNIST: <i>tissue type</i>	0.92±0.05	0.58±0.16	-	-
ICE-CD: <i>image type</i>	-	-	0.77±0.17	0.46±0.16
ICE-CD: <i>anatomical location</i>	-	-	0.75±0.22	0.37±0.20
Feature clustering	0.79±0.25	0.52±0.23	0.83±0.06	0.36±0.07
Gradient clustering	0.92±0.04	0.69±0.04	0.83±0.12	0.48±0.18

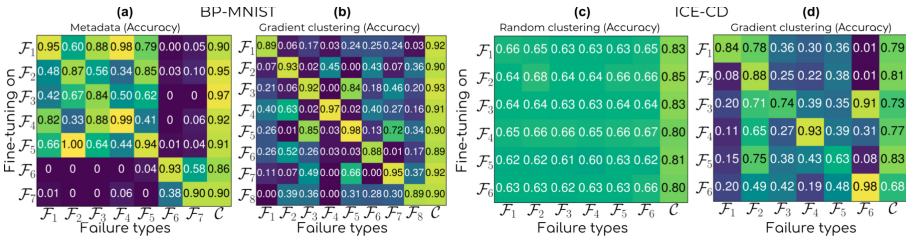


Fig. 2. Accuracy of fine-tuning on each failure type on BP-MNIST according to tissue types (a) and gradient clustering (b) and on ICE-CD according to random clustering (c) and gradient clustering (d). The lack of independence is apparent for tissue types (a) and random clustering (c) while a diagonal pattern is noticeable in (b) and (d).

4.2 Analysis of Automatically Discovered Failure Subtypes

We aim to inspect the automatically discovered subtypes on BP-MNIST and ICE-CD by *Gradient clustering* which achieved the best scores.

Binary PathMNIST (BP-MNIST): first, we observe that False positives and false negatives were mostly separated into two sets of clusters (i.e., each cluster contains mostly either circles or crosses as shown in Fig. 3(a)). Moreover, the two malignant tissue types were recovered separately: Cluster 2 and 4 for *cancer-associated stroma* and cluster 1 for *colorectal adenocarcinoma epithelium*. Secondly, even within one tissue type, *Gradient clustering* was able to discover independent failure types. Cluster 2 and 4 both focused on *cancer-associated stroma* but were relatively independent and differed when evaluating on \mathcal{F}_1 , \mathcal{F}_2 and \mathcal{F}_4 (See Fig. 2.). In addition, clusters 2 and 4 were visually different as seen in Fig. 3 with cluster 4 corresponding to darker less textured images. Finally, only cluster 8 seemed to contain *normal colon mucosa* (in addition to *Debris*) and does seem to contain darker textured images than other false positive clusters.

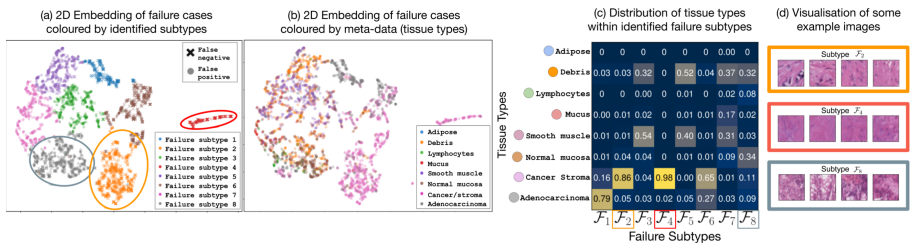


Fig. 3. Inspection of failure types obtained through gradient space clustering for BP-MNIST. (a) UMAP embedding coloured by *failure types*. (b) UMAP embedding coloured by *tissue types*. (c) Distribution of tissue types within each failure type. (d) Example images for remarkable failure types. Failures \mathcal{F}_2 and \mathcal{F}_4 which contain the same tissue but different visual appearances are identified as two different types.

Catheter detection (ICE-CD): *Gradient clustering* was able to recover a known and important but under-represented failure type: Cluster 4 (red cluster in Fig. 4(a)) focused on *Near-probe* catheters which are close to the ultrasound probe. Indeed, these catheters are hard to detect due to noise in this region of the images. Secondly, *Gradient clustering* was able to automatically discover some of the anatomical locations. Indeed, Cluster 6 (brown cluster in Fig. 4(a)) focused on the *LAA* and Cluster 3 (green cluster in Fig. 4(a)) focused mostly on the *SVC/IVC*. Finally, *Gradient clustering* was able to separate false positives (see the orange cluster in Fig. 4(a)) from false negatives (the others).

4.3 Model Repairment

Experiments: we aim to evaluate how much failure types performance can be improved while retaining performance on correct cases (See Table 2). We compare several repairment approaches on both datasets based on fine-tuning and EWC [11]. The fine-tuning is done on either a single failure type: \mathcal{F}_i^{tr} or all: $\mathcal{F} = \cup_i \mathcal{F}_i^{tr}$. Also, we compare to using \mathcal{C}^{tr} with a 50% sample ratio. For

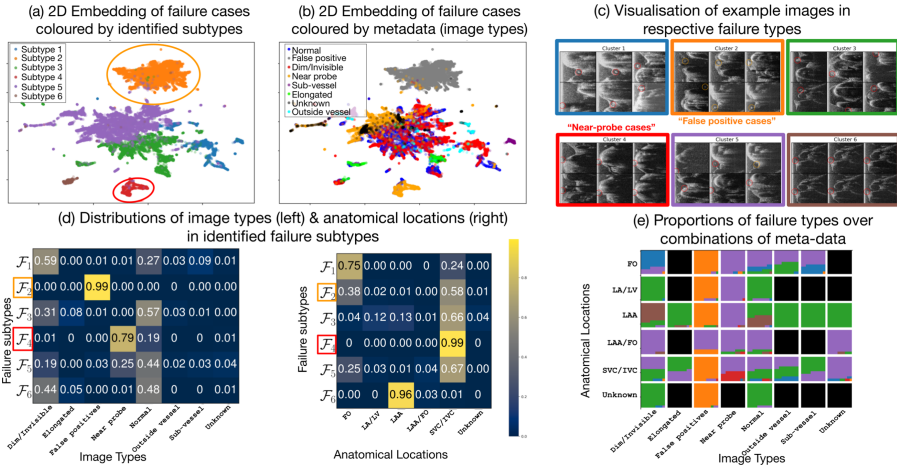


Fig. 4. Inspection of the failure types obtained through gradient space clustering for ICE-CD. UMAP embedding coloured by failure types (a) and by image types (b). (c) Example images for each subtype. (d) Distribution of image types and anatomical locations within each subtype. (e) Proportions of failure types over each combination of anatomical location and image types coloured by failure type. Two outstanding failure types are \mathcal{F}_2 focusing on “false positives” and \mathcal{F}_4 focusing on “Near-probe” cases.

all methods, early stopping is performed by selecting the best accuracy on \mathcal{F}_i^{tr} . Models were fine-tuned with a learning rate of 10^{-6} on BP-MNIST and 10^{-4} on ICE-CD and weight decay of 10^{-3} . Table 2 reports the accuracy on the test set of each failure type \mathcal{F}_i^{te} , the previously correct cases \mathcal{C}^{te} and the overall test set.

Analysis: first, fine-tuning on a single failure type \mathcal{F}_i^{tr} generalized more on that specific failure type than fine-tuning on all incorrect cases \mathcal{F}^{tr} at once (see Table 2). This may indicate that the failure types are conflicting during fine-tuning and it is more difficult to simultaneously learn a diverse set of cases than simple ones. Therefore, if learning unimportant failures is conflicting with critical ones, it makes sense to first start by repairing a carefully selected subset of the failures. Secondly, fine-tuning on the failures only couldn’t maintain performance on \mathcal{C}^{te} while including the correct cases \mathcal{C}^{tr} helped to preserve performance. Fine-tuning on $\mathcal{F}_i^{tr} \cup \mathcal{C}^{tr}$ for ICE-CD dropped to 0.73 but this was still higher than 0.32 if using only \mathcal{F}_i^{tr} for fine-tuning. Finally, while for EWC [11] the performance on correct cases didn’t drop as much as simple fine-tuning, EWC wasn’t able to maintain correct cases accuracy to more than 0.27 and 0.51.

Table 2. Comparison of model repairment methods on BP-MNIST (top) and ICE-CD (bottom) evaluated on failure types obtained in gradient space (ACC). Methods retaining at least 0.85 accuracy on \mathcal{C} are shown in green and those that do not in red.

Method (BP-MNIST)	\mathcal{F}_1^{te}	\mathcal{F}_2^{te}	\mathcal{F}_3^{te}	\mathcal{F}_4^{te}	\mathcal{F}_5^{te}	\mathcal{F}_6^{te}	\mathcal{F}_7^{te}	\mathcal{F}_8^{te}	\mathcal{C}^{te}	All
Pre-repairment	0	0	0	0	0	0	0	0	1.0	0.91
Fine-tuning on a single \mathcal{F}_i^{tr}	0.92	0.94	0.97	1.00	0.99	0.94	0.96	0.92	0.81	0.77
Fine-tuning on $\mathcal{F}_i^{tr} \cup \mathcal{C}^{tr}$	0.91	0.93	0.92	0.97	0.99	0.89	0.88	0.89	0.91	0.86
Fine-tuning on $\mathcal{F} = \cup_i \mathcal{F}_i^{tr}$	0.4	0.79	0.72	0.83	0.87	0.71	0.82	0.78	0.19	0.24
EWC [11] on \mathcal{F}^{tr}	0.34	0.65	0.79	0.79	0.85	0.64	0.79	0.76	0.27	0.30
Fine-tuning on $\mathcal{F}^{tr} \cup \mathcal{C}^{tr}$	0.25	0.65	0.71	0.66	0.80	0.48	0.69	0.68	0.9	0.88
Method (ICE-CD)	\mathcal{F}_1^{te}	\mathcal{F}_2^{te}	\mathcal{F}_3^{te}	\mathcal{F}_4^{te}	\mathcal{F}_5^{te}	\mathcal{F}_6^{te}	–	–	\mathcal{C}^{te}	All
Pre-repairment	0	0	0	0	0	0	–	–	1	0.59
Fine-tuning on a single \mathcal{F}_i^{tr}	0.85	0.99	0.77	0.69	0.64	0.96	–	–	0.32	0.36
Fine-tuning on $\mathcal{F}_i^{tr} \cup \mathcal{C}^{tr}$	0.81	0.97	0.79	0.96	0.67	0.97	–	–	0.73	0.64
Fine-tuning on $\mathcal{F} = \cup_i \mathcal{F}_i^{tr}$	0.61	0.9	0.71	0.28	0.57	0.72	–	–	0.64	0.66
EWC [11] on \mathcal{F}^{tr}	0.41	0.89	0.59	0.26	0.47	0.94	–	–	0.51	0.56
Fine-tuning $\mathcal{F}^{tr} \cup \mathcal{C}^{tr}$	0.41	0.8	0.65	0.39	0.61	0.82	–	–	0.86	0.78

5 Conclusion

We have introduced a principled framework to address the problems of failure identification, analysis and model repairment. Firstly, we put forward a set of desirable properties for meaningful failure types and novel surrogate metrics to assess those properties in practice. Secondly, we argued that model repairment should not only aim to fix the failures but also to retain performance on the previously correct data. Finally, we showed specific instantiations of our framework and demonstrated that clustering in feature and gradient space can automatically identify clinically important failures and outperform manual inspection.

References

1. Panfilov, E., Tiulpin, A., Klein, S., Nieminen, M.T., Saarakkala, S.: Improving robustness of deep learning based knee MRI segmentation: Mixup and adversarial domain adaptation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (2019)
2. Bdair, T., Navab, N., Albarqouni, S.: Roam: Random layer mixup for semi-supervised learning in medical imaging. arXiv preprint [arXiv:2003.09439](https://arxiv.org/abs/2003.09439) (2020)
3. Billot, B., Greve, D., Van Leemput, K., Fischl, B., Iglesias, J.E., Dalca, A.V.: A learning strategy for contrast-agnostic mri segmentation. arXiv preprint [arXiv:2003.01995](https://arxiv.org/abs/2003.01995) (2020)
4. Liu, Q., Dou, Q., Yu, L., Heng, P.A.: Ms-net: multi-site network for improving prostate segmentation with heterogeneous MRI data. IEEE Trans. Med. Imaging **39**(9), 2713–2724 (2020)
5. Dou, Q., de Castro, D.C., Kamnitsas, K., Glocker, B.: Domain generalization via model-agnostic learning of semantic features. Adv. Neural Inf. Process. Syst. **32**, 6450–6461 (2019)
6. Collins, G.S., Moons, K.G.M.: Reporting of artificial intelligence prediction models. Lancet **393**(10181), 1577–1579 (2019)

7. Liu, X., Rivera, S.C., Moher, D., Calvert, M.J., Denniston, A.K.: Reporting guidelines for clinical trial reports for interventions involving artificial intelligence: the consort-AI extension. *BMJ*, 370 (2020)
8. Oakden-Rayner, L., Dunnmon, J., Carneiro, G., Ré, C.: Hidden stratification causes clinically meaningful failures in machine learning for medical imaging. In: Proceedings of the ACM conference on health, inference, and learning, pp. 151–159 (2020)
9. Singla, S., Nushi, B., Shah, S., Kamar, E., Horvitz, E.: Understanding failures of deep networks via robust feature extraction. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2021)
10. McInnes, L., Healy, J., Saul, N., Grossberger, L.: UMAP: Uniform Manifold Approximation and Projection. *J. Open Source Softw.* **3**(29), 861 (2018)
11. Kirkpatrick, J., et al.: Overcoming catastrophic forgetting in neural networks. *Proc. Nat. Acad. Sci.* **114**(13), 3521–3526 (2017)
12. Karani, N., Chaitanya, K., Baumgartner, C., Konukoglu, E.: A lifelong learning approach to brain MR segmentation across scanners and protocols. In: Frangi, A.F., Schnabel, J.A., Davatzikos, C., Alberola-López, C., Fichtinger, G. (eds.) MICCAI 2018. LNCS, vol. 11070, pp. 476–484. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-00928-1_54
13. Hofmanninger, J., Perkonigg, M., Brink, J.A., Pianykh, O., Herold, C., Langs, G.: Dynamic memory to alleviate catastrophic forgetting in continuous learning settings. In: Martel, A.L., Abolmaesumi, P., Stoyanov, D., Mateus, D., Zuluaga, M.A., Zhou, S.K., Racoceanu, D., Joskowicz, L. (eds.) MICCAI 2020. LNCS, vol. 12262, pp. 359–368. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-59713-9_35
14. Karani, N., Erdil, E., Chaitanya, K., Konukoglu, E.: Test-time adaptable neural networks for robust medical image segmentation. *Med. Image Anal.* **68**, 101907 (2021)
15. Kamnitsas, K., et al.: Unsupervised domain adaptation in brain lesion segmentation with adversarial networks. In: Niethammer, M., et al. (eds.) IPMI 2017. LNCS, vol. 10265, pp. 597–609. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-59050-9_47
16. Yang, J., Shi, R., Ni, B.: Medmnist classification decathlon: a lightweight automl benchmark for medical image analysis. arXiv preprint [arXiv:2010.14925](https://arxiv.org/abs/2010.14925) (2020)
17. Kather, J.N., et al.: Predicting survival from colorectal cancer histology slides using deep learning: a retrospective multicenter study. *PLoS Med.* **16**(1), e1002730 (2019)
18. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint [arXiv:1409.1556](https://arxiv.org/abs/1409.1556) (2014)