Grand-Challenge.org: From Biomedical Image Analysis Challenges to Clinicians' Workflows

James A. Meakin **Bram van Ginneken**
Diagnostic Image Analysis Group
Radboud University Medical Center, Nijmegen, The Netherlands
{james.meakin|bram.vanginneken}@radboudumc.nl

1 Introduction

An increasing reliance on imaging for clinical decision making and an ageing population has resulted in an unprecedented demand for trained personnel to interpret medical images. As this demand is unable to be met there is consequently an increased workload for physicians, increasing the likelihood of interpretation errors [1]. Computer-aided detection and diagnosis systems have been developed to assist in reducing this workload and to enable more reliable clinical decisions. Challenges in biomedical image analysis have proven successful in driving innovation in creating algorithms that form components of these systems. However, for the algorithm developer, there exists a validation gap between performing well on the curated challenge data to transitioning to the clinic where imaging devices, scanning protocols and populations can vary significantly, and integration with heterogeneous clinical systems and clinicians' workflows is non-trivial.

Since 2012, as part of the Consortium for Open Medical Image Computing, we have developed and maintained an open source framework for hosting challenges in biomedical imaging, and support an instance of this at `https://grand-challenge.org`. Here, we have hosted successful challenges across various medical domains, such as LUNA16 for lung nodule detection [2], CAMELYON16/17 in digital pathology [3] and The Medical Imaging Decathlon for 10 segmentation tasks in CT and MRI [4]. The platform is now integral to our groups work, from private challenges used in educational courses to reproducibility of scientific output and archiving of developed algorithms. The platform has recently been extended to assist the algorithm developer in clinical validation by allowing clinicians to execute algorithms on their own data via a web interface.

2 Methods

The grand-challenge.org project is developed on GitHub and consists of 3 main components: 1. the grand-challenge.org framework, a re-usable platform for hosting challenges [5]. 2. Evalutils, a pip installable python package that assists challenge administrators in creating Docker containers for evaluating submissions from challenge participants [6]. 3. CIRRUS, a platform for developing medical imaging workstations that integrates with clinicians' workflows [7]. Whilst grand-challenge and evalutils are open source and licensed under Apache 2.0 and MIT respectively, CIRRUS is currently closed source but alternative viewers that integrate with the grand-challenge API could also be used, such as AMI [8] or Cornerstone [9].

2.1 The Grand-Challenge.org Framework

The platform is a web application that uses Django 2.0, backed by a PostgreSQL database and a Celery task queue with a Redis message broker. The application is distributed as a set of Docker containers, which are automatically published to Docker Hub after all tests have passed in Travis-CI. A Docker Compose file allows site administrators to quickly launch another instance of the framework on their own infrastructure, or for developers to replicate the entire stack on their machine.

Preprint. Work in progress.

We maintain an instance of the framework at `https://grand-challenge.org`. Here, anyone can join the site and become a challenge administrator by creating their own challenge. Other users are also able to join the site, sign up as participants in particular challenges and collaborate on solutions as teams. Currently, there are 17,000 users, participating in 101 public and private challenges hosted on the site. We also index challenges in medical imaging hosted on other sites, providing a valuable resource to the community in searching for datasets and benchmarks relevant to their tasks.

A challenge in the framework consists of a set of pages that describe the challenge, datasets for training and testing, the ground truth annotations for those datasets, and a method to evaluate submissions. Challenge participants submit their predictions as CSV files or ITK images. Each submission will be evaluated using that challenges evaluation method, which is provided by the challenge administrator as a Docker image. An instance of the evaluation image will be launched with the current submission attached as a Docker volume, and the container must produce a JSON file containing the score for this submission, which is then aggregated in the database. Instances of these images will be scheduled by Celery on an available Docker host, which for security reasons could be inside an isolated virtual machine.

Whilst challenge administrators are free to implement the evaluation in any language they choose, we have developed Evalutils, a Python 3.6+ package that assists the creation of evaluation containers. The package uses CookieCutter to generate project templates for Classification, Segmentation and Detection challenges, provides methods to validate submissions and give feedback to the participants, and to score evaluations using a statistics library designed for medical imaging tasks.

2.2 Reproducible Science

Researchers are now able to use the framework as a platform for reproducible science and algorithm archival. If the data can be shared publicly they are separately archived to Zenodo [10] where they receive a DOI or distributed via Academic Torrents [11] if the data are large. Those developing algorithms set their problem up as a challenge on grand-challenge.org and are expected to produce three containers: 1. a training container that trains the model and produces all the figures in the paper, 2. a model container that execute the model on new data and produce a prediction, 3. an evaluation container that will score predictions. Whilst the training container is archived and executed on our HPC systems, researchers are able to upload model containers to Grand Challenge in the same manner as for evaluation containers.

2.3 Integration into Annotation Workflows and Clinical Use

For clinical validation, these models need to be executed on diverse data from heterogeneous sources. Authorised algorithm users are now able to upload their own data and generate predictions from any the archived model containers available to them. Instances of these model containers are scheduled in a similar manner to the evaluation containers, only that the Nvidia Docker runtime is used and a GPU is attached to the container instance if required.

The predictions generated then need to be assessed in an environment the clinician is familiar with. Our group has previously developed CIRRUS for creating workstations, which is built on MeVisLab [12] and has been used as part of commercial CE and FDA certified products for Chest-CT screening [13, 14]. We have now extended the platform to make it accessible via a web browser so that clinicians are able to launch viewer instances wherever they are located, access algorithms and load any image and prediction available to them on Grand Challenge via a REST API. A clinician can then provide feedback on algorithm performance, generating more ground truth data by correcting the predictions, helping the algorithm developer close the validation gap.

3 Discussion

The next step for the framework is to accept model containers from challenge participants so that the test data can remain private. However, whilst challenge administrators are trusted, we do not consider challenge participants to be trusted users. This raises issues of how to run untrusted containers, for which we are investigating Singularity [15].

References

[1] C. S. Lee, P. G. Nagy, S. J. Weaver, and D. E. Newman-Toker, "Cognitive and system factors contributing to diagnostic errors in radiology," *American Journal of Roentgenology*, vol. 201, no. 3, pp. 611–617, Aug 2013. [Online]. Available: https://doi.org/10.2214/AJR.12.10375

[2] A. A. A. Setio, A. Traverso, T. de Bel, M. S. N. Berens, C. v. d. Bogaard, P. Cerello, H. Chen, Q. Dou, M. E. Fantacci, B. Geurts, R. v. d. Gugten, P. A. Heng, B. Jansen, M. M. J. de Kaste, V. Kotov, J. Y.-H. Lin, J. T. M. C. Manders, A. Sóñora-Mengana, J. C. García-Naranjo, E. Papavasileiou, M. Prokop, M. Saletta, C. M. Schaefer-Prokop, E. T. Scholten, L. Scholten, M. M. Snoeren, E. L. Torres, J. Vandemeulebroucke, N. Walasek, G. C. A. Zuidhof, B. v. Ginneken, and C. Jacobs, "Validation, comparison, and combination of algorithms for automatic detection of pulmonary nodules in computed tomography images: The LUNA16 challenge," *Medical Image Analysis*, vol. 42, pp. 1–13, 2017. [Online]. Available: https://arxiv.org/abs/1612.08012

[3] B. Ehteshami Bejnordi, M. Veta, P. J. van Diest, B. van Ginneken, N. Karssemeijer, G. Litjens, J. A. W. M. van der Laak, the CAMELYON16 Consortium, M. Hermsen, Q. F. Manson, M. Balkenhol, O. Geessink, N. Stathonikos, M. C. van Dijk, P. Bult, F. Beca, A. H. Beck, D. Wang, A. Khosla, R. Gargeya, H. Irshad, A. Zhong, Q. Dou, Q. Li, H. Chen, H.-J. Lin, P.-A. Heng, C. Haß, E. Bruni, Q. Wong, U. Halici, M. U. Öner, R. Cetin-Atalay, M. Berseth, V. Khvatkov, A. Vylegzhanin, O. Kraus, M. Shaban, N. Rajpoot, R. Awan, K. Sirinukunwattana, T. Qaiser, Y.-W. Tsang, D. Tellez, J. Annuscheit, P. Hufnagl, M. Valkonen, K. Kartasalo, L. Latonen, P. Ruusuvuori, K. Liimatainen, S. Albarqouni, B. Mungal, A. George, S. Demirci, N. Navab, S. Watanabe, S. Seno, Y. Takenaka, H. Matsuda, H. Ahmady Phoulady, V. Kovalev, A. Kalinovsky, V. Liauchuk, G. Bueno, M. M. Fernandez-Carrobles, I. Serrano, O. Deniz, D. Racoceanu, and R. Venâncio, "Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer," *Journal of the American Medical Association*, vol. 318, pp. 2199–2210, Dec. 2017.

[4] "The Medical Segmentation Decathlon," 2018. [Online]. Available: https://decathlon.grand-challenge.org/

[5] The Consortium for Open Medical Image Computing, "Grand-challenge.org," 2012. [Online]. Available: https://github.com/comic/grand-challenge.org

[6] ——, "Evalutils," 2018. [Online]. Available: https://github.com/comic/evalutils

[7] B. V. Ginneken, E. M. V. Rikxoort, S. J. Lafebre, C. Jacobs, M. Schmidt, J.-M. Kuhnigk, M. Prokop, C. M. Schaefer-Prokop, J.-P. Charbonnier, L. Hogeweg, P. Maduskar, L. G. Estrella, R. Philipsen, and B. C. Lassen, "CIRRUS lung: An optimized workflow for quantitative image analysis of thoracic computed tomography and chest radiography for major pulmonary diseases—chronic obstructive pulmonary disease, lung cancer and tuberculosis," in *RSNA Quantitative Imaging Reading Room Showcase*, Chicago, USA, 2013.

[8] Fetal-Neonatal Neuroimaging Developmental Science Center, Boston Childrens Hospital, "AMI Medical Imaging Javascript ToolKit," 2016. [Online]. Available: https://github.com/FNNDSC/ami

[9] "Cornerstone JS," 2014. [Online]. Available: https://github.com/cornerstonejs/cornerstone

[10] "Zenodo," 2018. [Online]. Available: https://zenodo.org/

[11] H. Z. Lo and J. P. Cohen, "Academic torrents: Scalable data distribution," *CoRR*, vol. abs/1603.04395, 2016. [Online]. Available: http://arxiv.org/abs/1603.04395

[12] F. Ritter, T. Boskamp, A. Homeyer, H. Laue, M. Schwier, F. Link, and H. . Peitgen, "Medical image analysis," *IEEE Pulse*, vol. 2, no. 6, pp. 60–70, Nov 2011.

[13] MeVis Medical Solutions AG, "Veolity," 2014. [Online]. Available: https://www.veolity.com/

[14] Invivo, "DynaCAD lung," 2014. [Online]. Available: http://www.invivocorp.com/solutions/lung-cancer-screening/

[15] G. M. Kurtzer, V. Sochat, and M. W. Bauer, "Singularity: Scientific containers for mobility of compute," *PLOS ONE*, vol. 12, no. 5, pp. 1–20, 05 2017. [Online]. Available: https://doi.org/10.1371/journal.pone.0177459