



Deploying automated machine learning for computer vision projects: a brief introduction for endoscopists

Neal Mahajan, ScB,¹ Erik Holzwanger, MD,¹ Jeremy Glissen Brown, MD,² Tyler M. Berzin, MD³

INTRODUCTION

Artificial intelligence (AI) and machine learning (ML) will play a growing role in gastroenterology. Computer-aided polyp detection is an early example of the how AI/ML can be applied to clinical practice.¹ There is a broad array of potential applications for ML and computer vision in GI endoscopy. The most powerful computer vision models today use advanced deep learning algorithms that may be unfamiliar to most gastroenterologists, given that courses in machine learning have not yet been incorporated into most medical training pathways.² Automated machine learning (AutoML) may be a useful tool to bridge this divide. AutoML platforms aim to simplify the ML pathway by automatically determining the best model for the provided data and task.³ This is done through automatic hyperparameter tuning. Hyperparameters are user-specified settings that control the model training process. In traditional ML, hyperparameters must be selected by a computer scientist using previous knowledge or a systematic search strategy. AutoML implements systematic search strategies during training allowing for model configurations that can rival and surpass those of human experts.⁴ While some basic coding skills are still required, a physician with rudimentary Python (Python Software Foundation, Wilmington, Del, USA) skills and a labeled dataset can use AutoML to build a model to detect or diagnose a variety of endoscopic findings. This project demonstrates how computer-aided detection (CADe) models can be created using AutoML on publicly available datasets for colon polyps. CADe is used to detect an object and provide its

Abbreviations: AI, artificial intelligence; AutoML, automated machine learning; CADe, computer-aided detection; CADx, computer-aided diagnosis; IoU, intersection over union; ML, machine learning.

Copyright © 2023 American Society for Gastrointestinal Endoscopy. Published by Elsevier Inc. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).
2468-4481
<https://doi.org/10.1016/j.vgie.2023.02.012>

Center for Advanced Endoscopy, Division of Gastroenterology, Beth Israel Deaconess Medical Center and Harvard Medical School, Boston, Massachusetts (1), Division of Gastroenterology, Duke University Medical Center, Durham, North Carolina (2), Center for Advanced Endoscopy, Division of Gastroenterology, Beth Israel Deaconess Medical Center and Harvard Medical School, Boston, Massachusetts (3).

location to the user using a bounding box that frames the object.¹

THE SUN DATASET

The SUN Dataset (<https://groups.csail.mit.edu/vision/SUN/hierarchy.html>) contains image frames from 100 colonoscopies with over 150,000 annotated polyp and nonpolyp images. The annotation files for the data contain the bounding box, a set of coordinates that form a rectangle that encloses the lesion on an image, and pathology of each polyp.⁵ The Google AutoML platform (Google, Mountain View, Calif, USA) requires a specific annotation format (<https://cloud.google.com/vertex-ai/docs/datasets/prepare-image>), so the annotation file needs to be reformatted using a Python script for minimal parsing. The Python script is simply a few lines of code that alter the format of the annotations for each line in the annotation file. This enables the annotation file to be readable by the platform being tested. The codes for the parsing are available here: Python parsing code, <https://github.com/nealmahajan/Google-AutoML-Endoscopy>; SUN Database, <http://sundatabase.org/>; Python and Numpy Tutorial, <https://cs231n.github.io/python-numpy-tutorial/>.

USING GOOGLE AUTOML

To use Google AutoML, it is first necessary to create a Google cloud account and a Google storage bucket, which is similar to a file folder to store data (Video 1, available online at www.videogie.org). Next, the images can be uploaded using the Google Cloud CLI software development kit (<https://cloud.google.com/sdk/docs/install>). After using the Python script to generate the import file, the import file is uploaded to the AutoML platform to associate the correct label with each image and generate the dataset that will be used for the model training.

After successful importation of the data into the AutoML platform, we then set the parameters for the model we want to create, including whether we want to maximize the model accuracy or to have minimal latency. Additionally, the user specifies the maximum number of node hours they would like to use for model training. Node hours are a measure of computer resources. A reasonable

True label	Predicted label	serrated_adenoma	invasive_cancer	sessile_serrated_lesion	adenoma	not-polyp	hyperplastic
serrated_adenoma	97%	—	—	1%	2%	—	—
invasive_cancer	—	94%	—	1%	5%	—	—
sessile_serrated_lesion	—	—	87%	2%	11%	—	—
adenoma	—	—	—	96%	4%	—	—
not-polyp	—	—	—	1%	99%	—	—
hyperplastic	—	—	—	8%	8%	85%	—

Figure 1. Confusion matrix for Google AutoML computer-aided diagnosis model.

starting point for this value is 20 node hours for object detection and 15 node hours for image classification. The model will train on Google's servers and an email will be sent to your account once training has finished.

RESULTS

The CADe model trained in Google AutoML had an average precision of 0.789. The precision at an intersection over union (IoU) threshold of 0.3 is 0.977 compared to 0.997 for a purpose-built model based on the SUN Dataset. In general, the higher the IoU threshold, the lower the precision because the predicted and actual bounding box need to overlap more for it to be considered a true positive. The average precision in this context is the average of all the precisions when the model is evaluated across all IoU thresholds. The platform creates precision recall curves for the model, which illustrate the tradeoffs between eliminating false positives and false negatives. The curve helps the user find the confidence threshold that will maximize precision and recall. For this model, the default confident threshold of 0.5 does not yield the best tradeoff, so we manually adjust it to 0.32. The F1 score, a measure of accuracy that balances precision and recall, for the default confidence threshold of 0.5 is 0.94. The F1 score for the user-defined confidence threshold of 0.32 is 0.965. The interface provides examples of false negatives, false positives, and true positives, which can be used to improve the model. For example, if there are

many false negatives or positives caused by a particular polyp morphology or normal anatomic feature, more images from that category could be added to the dataset to improve the model.

In addition to object *detection*, Google AutoML also allows you to make image *classification* models that can be used to predict pathology, known as computer-aided diagnosis (CADx). The confusion matrix for the CADx model indicates how often the model predicts the correct label and the most common incorrect label predictions for each class (Fig. 1).

CONCLUSIONS

Google AutoML reduces the difficulty of creating image detection and classification models from data by using automatic hyperparameter tuning and an intuitive user interface, as well as by removing the burden of using and configuring a graphics processing unit. Here we demonstrate the steps required for a gastroenterologist to train CADe models using a publicly available dataset of polyp images. However, improvements can be made by accepting different data annotation formats and providing more information about the final model. Google AutoML can be a useful tool for rapidly prototyping computer vision models for gastroenterologists and may be a useful entry point as a project to learn basic ML skills which can be applied to computer vision projects in GI endoscopy. While the AutoML tools reviewed here are currently only useable on still images, future iterations

of these technologies will allow users to develop and deploy similar tools on live video.

ACKNOWLEDGMENTS

The authors thank Dr Sravanthi Parasa for her review and expertise.

DISCLOSURE

Dr Holzwanger is a medical advisor for Virgo. Dr Glissen Brown is a scientific advisor for Medtronic on projects applying artificial intelligence to GI endoscopy. Dr Berzin is a consultant for Medtronic, Wision AI, Docbot AI, and Magentiq Eye. Mr Mahajan disclosed no financial relationships.

REFERENCES

1. Glissen Brown JR, Mansour NM, Wang P, et al. Deep learning computer-aided polyp detection reduces adenoma miss rate: a United States multi-center randomized tandem colonoscopy study (CADET-CS Trial). *Clin Gastroenterol Hepatol* 2022;20:1499-507.e4.
2. Chai J, Zeng H, Li A, et al. Deep learning in computer vision: a critical review of emerging techniques and application scenarios. *Mach Learn Appl* 2021;6:100134.
3. Waring J, Lindvall C, Umeton R. Automated machine learning: review of the state-of-the-art and opportunities for healthcare. *Artif Intell Med* 2020;104:101822.
4. Komer B, Bergstra J, Eliasmith C. Hyperopt-sklearn: automatic hyperparameter configuration for scikit-learn. In: *Proceedings of the 13th Python in Science Conference: SciPy*; 2014. Available at: <https://conference.scipy.org/proceedings/scipy2014/pdfs/komer.pdf>. Accessed April 4, 2023.
5. Misawa M, Kudo S, Mori Y, et al. Development of a computer-aided detection system for colonoscopy and a publicly accessible large colonoscopy video database (with video). *Gastrointest Endosc* 2021;93:960-7.e3.

Endoscopedia

Endoscopedia has a new look! Check out the redesign of the official blog of *GIE* and *VideoGIE*. Keep up with the latest news and article discussions and post your comments or questions to *VideoGIE* authors. Visit us at www.endoscopedia.com.