Transfer learning with CNN architectures for classifying gastrointestinal diseases and anatomical landmarks

Danielle Dias, Ulisses Dias
University of Campinas, Brazil
danielle.dias@ic.unicamp.br,ulisses@ft.unicamp.br

ABSTRACT

Transfer learning is an approach where a model trained for a given task is used as a starting point on a second task. Many advanced deep learning architectures have been pre-trained on ImageNet and are currently available, which makes this technique very popular. We evaluate 10 pre-trained architectures on the task of finding gastrointestinal diseases and anatomical landmarks in images collected in hospitals. Our analysis considered both processing time and accuracy. We also study if global image features bring advantages to the pre-trained models for the problem of gastrointestinal medical image classification. Our best models achieved accuracy and F1-score values of 0.988 and 0.908, respectively. Our fastest model classifies an input instance in 0.037 seconds, and yields accuracy and F1-score of 0.983 and 0.866, respectively.

1 INTRODUCTION

The Medico Task proposes the challenge of predicting diseases based on multimedia data collected in hospitals [7, 8]. The images are frames collected from videos captured by the insertion of a camera in the gastrointestinal tract. The main purpose is to identify anomalies that can be detected visually, even before they become symptomatic. More details can be found in the task overview [9].

To solve the task we created several models based on features extracted from deep convolutional architectures and global image features. The deep architectures were trained on ImageNet. The strategy was to create three kinds of models: (i) those having only features extracted from deep architectures as input, (ii) those that considered only global image features, and (iii) those created with all features available.

The approach of extracting features from pre-trained models is usually referred to as transfer learning. It has become popular because many models are available. We selected 10 architectures trained on ImageNet as extractors, and compared their performance on classifying images for the Medico Task.

The architectures differ in several characteristics, which impacts the time to compute the features. Since efficiency is an important matter on this task, we computed the data processing speed for each test images and made efforts to reach a balance between solution quality and running time.

2 RELATED WORK

Transfer learning has been used in several problems in a number of domains. In 2017, Agrawal *et al.* [1] used transfer learning for the Medico Task when the challenge had 8 classes and achieved good results. They restricted the analysis to two architectures: InceptionV3, and VGGNet. They also conducted an analysis of how a model performs if it uses features extracted from both architectures as input. The results are better for just a small factor, which lead us believe that it does not worth the extra processing time. We extended the work of Agrawal *et al.* [1] by using 10 architectures and by considering both solution quality and efficiency in our analysis.

MediaEval'18, 29-31 October 2018, Sophia Antipolis, France

3 APPROACH

The development and test dataset contain 5,293 and 8,740 images, respectively. For each image, visual features were extracted and provided as feature vectors by the task organizers, namely: JCD, Tamura, ColorLayout, EdgeHistogram, AutoColorCorrelogram and PHOG [6]. These feature vectors are sequences of floating point values for each image, and the number of values sum to 1185. These values were joined to form a table used as input to our model, where rows represent images. We removed 19 columns because either they had the same value for all images or because they were duplicated.

We used 10 architectures trained on ImageNet: DenseNet121 [5], DenseNet169 [5], DenseNet201 [5], InceptionResNetV2 [11], InceptionV3 [12], MobileNet [3], ResNet [4], VGG16 [10], VGG19 [10], Xception [2]. Each architecture requires a particular pre-processing step and returns vectors of floating point numbers. Vector sizes are: DenseNet121 (1024), DenseNet169 (1664), DenseNet201 (1920), InceptionResNetV2 (1536), InceptionV3 (2048), MobileNet (1024), ResNet (2048), VGG16 (512), VGG19 (512), Xception (2018).

The input layer has the same number of nodes as the feature vector sizes. That said, in a model that uses only global features the input layer has 1166 nodes. In a model that uses global features and a given architecture as feature extractor, the input layer has the feature vector of that architecture plus 1166.

The best model for most of the input features uses one hidden layer that has 512 nodes and each node uses Relu as activation function. We added a Dropout of 50% in the training stage and l_2 regularization to prevent overfit. Models with more layers tend to overfit very easily with just a few epochs. It would be possible to create simpler models for small feature vectors like VGG architectures, but we decided to report the same network for all input vectors for comparison purposes. The output layer has 16 nodes (one for each class) and uses softmax activation to classify the image in one of the classes.

4 RESULTS AND ANALYSIS

During the training stage we split the development set with 5,293 images into train (3,038 images), validation (1,722 images), and test (531 images) datasets. We used train and validation sets to train and

tune the classifier. Test dataset was used only once to generate the results in Table 1 after we were satisfied with the validation scores.

The model that uses only global image features yields an accuracy of 0.813, and F1-score of 0.782, which we consider a baseline result. Table 1 summarizes the results using transfer learning in the test dataset with 531 images. We based our decision about which model was best to submit on the F1-score and accuracy metrics.

Table 1: F1-score and accuracy of the transfer learning models in our test dataset (531 images). Table also reports the average time (in seconds) to classify an image after the model is loaded in memory. We highlighted the best results.

Architecture	No Global Features			Global Features	
	Time	ACC	F1	ACC	F1
DenseNet121	0.163	0.904	0.856	0.915	0.868
DenseNet169	0.209	0.915	0.899	0.919	**0.903**
DenseNet201	0.242	**0.923**	0.905	**0.925**	0.871
InceptResnetV2	0.349	0.908	0.894	0.904	0.858
InceptionV3	0.213	0.883	0.876	0.889	0.845
Mobilenet	**0.037**	0.889	0.841	0.896	0.850
Resnet	0.115	0.909	**0.916**	0.894	0.883
VGG16	0.372	0.879	0.837	0.870	0.828
VGG19	0.406	0.881	0.867	0.866	0.858
Xception	0.257	0.877	0.835	0.898	0.855

Our first decision was to disregard the models that had transfer learning plus global features because the improvement after adding global features was considered irrelevant. As an example, the architecture DenseNet201 had a small increase of accuracy from 0.923 to 0.925. These were the best models using accuracy metric. If we consider the F1-score, models with global image features became even worse in several cases. Taking into account that these models with global features have 1166 more inputs than their counterparts, we decided to keep the simpler models.

DenseNet201 and Resnet were selected as the two best models considering accuracy and F1-score, respectively. MobileNet was selected because it is amazingly fast, and efficiency is an important matter on this task. Indeed, we consider this model the best trade-off since its 0.889 of accuracy is not far away from the 0.923 accuracy of DenseNet201 (best accuracy model) and runs 6.5 times faster. DenseNet121 was the last model selected because is somewhere in between the best accuracy model (DenseNet201) and the fastest model (Mobilenet).

Selected models were submited and evaluated against the dataset with 8,740 images. In this dataset, our results were much better than we anticipated. Results are shown in Table 2, where we also report the official competition ranking indicator R_k. ResNet and DenseNet models achieved accuracy higher than 0.987. Mobilenet yields accuracy of 0.983, which is very close to the top accuracy of 0.988 achieved by DenseNet201 and Resnet.

F1-score shows that Resnet and DenseNet201 are the best models and that MobileNet is somewhat worse than the others. However, we believe MobileNet has best trade-off if we consider that it returns solution in much less than a second. DenseNet121 does not appear a good choice because it is slower than Resnet and present somewhat worse results. Therefore, Resnet should be preferred in any situation. DenseNet201 was the top accuracy and F1-score, but it is so close to Resnet that it is difficult to argue that it is enough compensation for the fact of being twice as slower.

Table 2: F1-score, accuracy and R_k of the selected models in task test dataset (8,740 images).

Architecture	ACC	F1	R_k
DenseNet121	0.987	0.903	0.893
DenseNet201	0.988	0.908	0.898
Mobilenet	0.983	0.866	0.853
Resnet	0.988	0.906	0.896

5 DISCUSSION AND OUTLOOK

We believe these models can be improved and the confusion matrix shown in Figure 1 provides some insights. We analysed how they performed in each of 16 class and found out that the class "out-of-patient" is particularly problematic, since it has only 4 instances in the development set and 5 instances in the test set. Furthermore, none of our models was able to classify these 5 instances right, which does not impact accuracy but degrades F1-score. In the future, some data augmentation should be performed to improve this class.

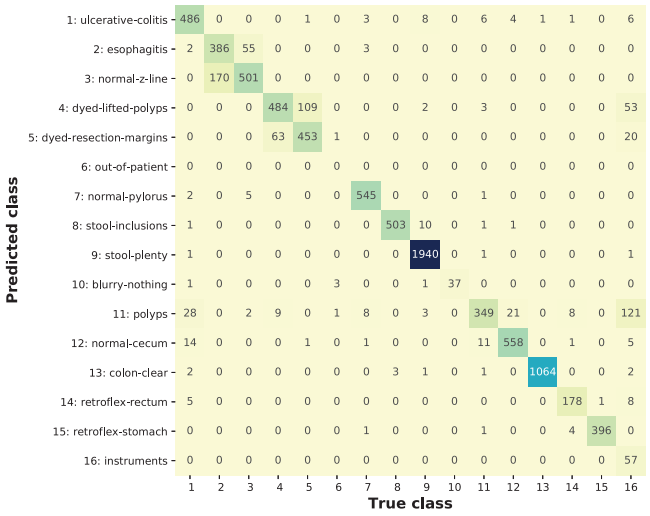

Figure 1: Confusion matrix of DenseNet201 on the test dataset with 8,740 images.

Another class we need to study is "esophagitis", because 170 instances were classified as "normal-z-line", which accounts for 30.57% of instances in this class. The classes "Stool-plenty" and "colon-clear" have most of the instances, and our models did a good job on classifying them right, which boosted the scores.

ACKNOWLEDGMENTS

We thank CAPES and CNPq (grant 400487/2016-0).

REFERENCES

[1] Taruna Agrawal, Rahul Gupta, Saurabh Sahu, and Carol Y Espy-Wilson. 2017. SCL-UMD at the Medico Task-MediaEval 2017: Transfer Learning based Classification of Medical Images.. In *Proceedings of the MediaEval 2017 Workshop*.

[2] François Chollet. 2016. Xception: Deep Learning with Depthwise Separable Convolutions. *CoRR* abs/1610.02357 (2016). arXiv:1610.02357 http://arxiv.org/abs/1610.02357

[3] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. 2017. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. (04 2017).

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Deep Residual Learning for Image Recognition. *CoRR* abs/1512.03385 (2015). arXiv:1512.03385 http://arxiv.org/abs/1512.03385

[5] Gao Huang, Zhuang Liu, and Kilian Q. Weinberger. 2016. Densely Connected Convolutional Networks. *CoRR* abs/1608.06993 (2016). arXiv:1608.06993 http://arxiv.org/abs/1608.06993

[6] Mathias Lux and Savvas A. Chatzichristofis. 2008. Lire: Lucene Image Retrieval: An Extensible Java CBIR Library. In *Proceedings of the 16th ACM International Conference on Multimedia (MM '08)*. ACM, New York, NY, USA, 1085–1088.

[7] Konstantin Pogorelov, Kristin Ranheim Randel, Thomas de Lange, Sigrun Losada Eskeland, Carsten Griwodz, Dag Johansen, Concetto Spampinato, Mario Taschwer, Mathias Lux, Peter Thelin Schmidt, Michael Riegler, and Pål Halvorsen. 2017. Nerthus: A Bowel Preparation Quality Video Dataset. In *Proceedings of the 8th ACM on Multimedia Systems Conference (MMSys'17)*. ACM, New York, NY, USA, 170–174. https://doi.org/10.1145/3083187.3083216

[8] Konstantin Pogorelov, Kristin Ranheim Randel, Carsten Griwodz, Sigrun Losada Eskeland, Thomas de Lange, Dag Johansen, Concetto Spampinato, Duc-Tien Dang-Nguyen, Mathias Lux, Peter Thelin Schmidt, Michael Riegler, and Pål Halvorsen. 2017. KVASIR: A Multi-Class Image Dataset for Computer Aided Gastrointestinal Disease Detection. In *Proceedings of the 8th ACM on Multimedia Systems Conference (MMSys'17)*. ACM, New York, NY, USA, 164–169. https://doi.org/10.1145/3083187.3083212

[9] Konstantin Pogorelov, Michael Riegler, Pål Halvorsen, Thomas de Lange, Kristin Ranheim Randel, Duc-Tien Dang-Nguyen, Mathias Lux, and Olga Ostroukhova. 2018. Medico Multimedia Task at MediaEval 2018. In *Proceedings of the MediaEval 2018 Workshop*.

[10] Karen Simonyan and Andrew Zisserman. 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition. *CoRR* abs/1409.1556 (2014). arXiv:1409.1556 http://arxiv.org/abs/1409.1556

[11] Christian Szegedy, Sergey Ioffe, and Vincent Vanhoucke. 2016. Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning. *CoRR* abs/1602.07261 (2016). arXiv:1602.07261 http://arxiv.org/abs/1602.07261

[12] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. 2015. Rethinking the Inception Architecture for Computer Vision. *CoRR* abs/1512.00567 (2015). arXiv:1512.00567 http://arxiv.org/abs/1512.00567