Gastrointestinal disease detection using ResNet on Hyper-kvasir dataset

Dev Patel
Btech Computer Science
(Ahmedabad Universirty)
dev.p1@ahduni.edu.in

Meet Patel
Btech Computer Science
(Ahmedabad Universirty)
meet.p6@ahudni.edu.in

Manthan Patel
Btech Computer Science
(Ahmedabad Universirty)
manthan.p@ahduni.edu.in

Sagar Bajaj
Btech Computer Science
(Ahmedabad Universirty)
sagar.b2@ahduni.edu.in

***Abstract*— Gastrointestinal disease detection is an important task in medical image analysis, and the Hyper-Kvasir dataset is a widely-used benchmark dataset for this task. ResNet (Residual Network) is a deep learning architecture that has achieved state-of-the-art performance on many image classification tasks. This report presents an implementation of ResNet50, a deep learning model, for the detection of gastrointestinal diseases using the hyper-kvasir dataset. The dataset contains 8,000 high-resolution images of the gastrointestinal tract, labeled with 23 different classes, and we have selected four classes for this project. We performed data preprocessing, including resizing images and data augmentation, followed by fine-tuning the pre-trained ResNet50 model and training the model using the Adam optimizer. The model's performance was evaluated on the testing set, and the results showed an accuracy of around 92% after five epochs. Overall, the implementation of ResNet50 on the hyper-kvasir dataset shows promising results in detecting gastrointestinal diseases, and further improvements can be made by optimizing the hyperparameters and experimenting with other deep-learning architectures.**

Keywords*— *Gastrointestinal diseases, Hyper-kvasir, deep learning, Residual Network (ResNet), fine-tuning.

I. Introduction

A collection of illnesses known as gastrointestinal (GI) diseases include those that affect the esophagus, stomach, small and large intestines, liver, gallbladder, and pancreas. Effective treatment and management of many disorders depend on an early and precise diagnosis. Deep learning-based methodologies have recently demonstrated promising outcomes in the identification and categorization of a variety of diseases. The deep learning model ResNet has been applied to a variety of computer vision tasks, including the processing of medical images. The Hyper-kvasir dataset may be used to train a ResNet model, which can then be used to correctly classify GI illnesses from endoscopic pictures. Better patient outcomes may result from increasing the effectiveness and precision of diagnosis and treatment. Deep neural networks suffer from the problem of vanishing gradients making it more difficult to train. As the network depth increases, the gradient multiplication during the backpropagation results in extremely small gradients. This results in stagnated or degraded performance by the network. ResNet is a deep convolution neural network architecture that proposed a residual learning framework for solving the degradation problem. ResNet-50 architecture includes 48 convolution layers, 1 max pooling, and 1 average pooling layer.

II. Literature Survey

Deeper neural network stack multiple layers which inculcate the problem of vanishing gradients during the process of backpropagation. When the gradients backpropagate using the chain rule, the repetitive multiplication of weights leads to extremely small weights resulting in stagnation or degradation of the performance of the neural network. Residual networks put forward the framework of skip connections through which the problem of degradation can be mitigated. The skip connection helps the deeper layer to learn an identity function that outputs the same result as the shallow layer and hence stacking multiple layers in between the connection does not affect the output. The original ResNet architecture that was proposed was ResNet-34 comprising 34 layers. The ResNet-50 was based on the original architecture with 50 layers that implemented the bottleneck design. The bottleneck block uses 1*1 convolutions which helps in reducing the number of parameters and matrix multiplication. This enables faster training associated with each layer. The ResNet-50 uses a stack of three layers and is divided into 4 main segments. It consists of 7*7,64 kernels with a stride of 2 and a max pooling layer with a stride of 2. The internal architecture comprises 48 layers with residual connections. Each convolution layer is followed by a batch normalization layer and a ReLU(rectified linear unit) activation function. The output is then convolved with an average pooling layer that is fully connected with 1024 neurons using the SoftMax activation function.

III. Implementation

The dataset used is the Hyper-kvasir dataset. The hyper-kvasir dataset contains 8,000 high-resolution images of the gastrointestinal tract, with each image labeled according to the type of disease present. There are two types of images. First is anatomical landmarks which represent a normal condition and are used for positioning during endoscopic examination. Other is pathological findings, which represent abnormality in the gastrointestinal tract. There are a total of 23 different classes in the dataset. We have selected 4 classes of pathological findings: Barrett's, Esophagitis, Polyps, and Ulcerative-colitis.

Disease prediction using the hyper-kvasir dataset involves the use of convolutional neural networks (CNN) to predict different gastrointestinal diseases. In this project, we will use the ResNet50 architecture, a deep-learning model that has proven effective in image classification tasks. To implement this project, we follow the following steps:

Data preprocessing: This step involves loading the hyper-kvasir dataset, resizing the images to a standard size, and dividing the dataset into training and testing sets. First, we split the dataset into training, validation, and test sets with 60% train set, 20% validation set, and 20% test set using the “split_folders” library. We resized all the images to the dimension (224,224) because the ResNet50 model requires images to be in that dimensions. We also applied data augmentation techniques such as random rotation, flipping, and zooming to increase the dataset's size and prevent overfitting. Image augmentation is performed using ImageDataGenerator, available in the Keras library. We performed the following data augmentation:

- shear_range = 0.2
- zoom_range = 0.2
- horizontal_flip = True
- batch_size = 32 (for training and validation) and 1 (for testing)

Model architecture: We compared different CNN architectures to decide which architecture to use for disease detection on the Hyper-kvasir dataset. We compared VGG16, ResNet50, DenseNet169, InceptionV3, EfficientNet-B0 and a 14-layer basic CNN model. We executed the models for 20 epochs and compared the accuracies and losses of the models. Following are the performance of the models on the Hyper-Kvasir image dataset:

Model	**Loss**	**Accuracy**
CNN	0.37	0.895
VGG-16	0.33	0.938
Inception-V3	0.81	0.928
DenseNet-169	0.62	0.94
Efficient Net-B0	0.58	0.812
ResNet-50	0.27	0.924

The ResNet50 model gives lowest train and validation losses. We will use the ResNet50 architecture, a pre-trained model in the Keras library. This architecture is composed of 50 layers and has skip connections, which help avoid the vanishing gradient problem and improve the model's accuracy.

Fine-tuning: In this step, we fine-tune the ResNet50 architecture by retraining the last few layers of the model. We freeze the initial layers of the model and only train the final layers to adapt the model to our specific classification task. Using the Keras library, we used the pre-trained ResNet50 model trained on the ImageNet dataset. We used the GlobalAveragePooling2D() method, which takes a 4D tensor as input and returns a 2D tensor with the same number of channels (depth) as the input tensor, but with spatial dimensions collapsed to a single dimension. We added a fully connected layer with 1024 neurons and a ReLU activation function on top of the pre-trained ResNet50 model. We defined the output layer with the Softmax activation function with 4 classes.

Training: We train the model using the training set, which involves feeding the preprocessed images to the ResNet50 architecture and updating the model's weights using backpropagation. We used categorical cross-entropy as the loss function and the Adam optimizer to update the model's weights.

Model evaluation: We evaluated the trained model using the testing set, which involves feeding the preprocessed images to the model and predicting the disease type for each image.

Disease detection on video dataset: We implemented video classification to detect a disease from the given video. We used the labeled videos from the Hyper-Kvasir dataset. There are 73 videos of only Polyps disease from a total 373 videos. There are very less number of videos of the other classes than Polyps. Therefore we did binary classification by classifying the given video into Polyps and healthy videos. There are 10 videos of Polyps and 18 videos of healthy conditions. The healthy class consists of videos from different classes of healthy conditions. To implement the video classification the first step is feature extraction. The videos are first we pre-process the video to convert the frames in the proper size by implementing a function to pre-process the videos. Then the ResNet50 model will extract the features from the videos and store the maximum of them in the features array. The correct labels will be stored in the labels array. The ResNet50 model used for feature extraction is the model we trained on our image dataset to classify the images. After doing feature extraction, we split the features and labels into 70% train set and 30 % test set. Then we trained a Multi-Layer Perceptron (MLP) classifier to classify the video based on the features extracted. We evaluated the MLP classifier on the 30% test set.

IV. Results

We trained the ResNet50 model on the image dataset for 30 epochs and the results are:

On training set:
Accuracy = 0.981
Loss = 0.0437

On validation set:
Accuracy = 0.9212
Loss = 0.259

On test set:
Accuracy = 0.9242
Loss = 0.2759

On the video dataset, the model gave an accuracy of 0.5 on test set and 0.682 on the train set. Here the MLP classifier is overfitting on the training set. So, we performed hyperparameter tuning on the MLP classifier using grid search. We tuned the hyperparameters such as activation function, alpha, hidden layers size, learning rate, maximum iterations, and optimizer. After hyperparameter tuning the best test accuracy we got is 0.684.

V. Conclusions

In conclusion, this report presented an implementation of the ResNet50 deep learning architecture for the detection of gastrointestinal diseases using the Hyper-Kvasir dataset. The dataset contained 8,000 high-resolution images labeled with 23 different classes, and we focused on four classes of pathological findings. The implementation included data preprocessing, such as resizing images and data augmentation, followed by fine-tuning the pre-trained ResNet50 model and training the model using the Adam optimizer. The model's performance was evaluated on the testing set, and the results showed an accuracy of around 92%. The study demonstrates that the ResNet50 model shows promising results in detecting gastrointestinal diseases, and further improvements can be made by optimizing the hyperparameters and experimenting with other deep-learning architectures. Overall, the study suggests that deep learning models can play a significant role in the early and precise diagnosis of GI diseases, leading to better patient outcomes.

References

[1] G. Eason, B. Noble, and I. N. Sneddon, “On certain integrals of Lipschitz-Hankel type involving products of Bessel functions,” Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955. *(references)*

[2] S. (n.d.). *GitHub - simula/hyper-kvasir: GitHub repository for the Hyper-Kvasir dataset.* GitHub. https://github.com/simula/hyper-kvasir

[3] Borgli, H., Thambawita, V., Smedsrud, P. H., Hicks, S. A., Kwon, G., Eskeland, S. L., Randel, K. R., Pogorelov, K., Lux, M., Nguyen, D. K., Johansen, D., Griwodz, C., Stensland, H. K., Garcia-Ceja, E., Schmidt, P., Hammer, H. L., Riegler, M., Halvorsen, P., & De Lange, T. (2020). HyperKvasir, a comprehensive multi-class image and video dataset for gastrointestinal endoscopy. *Scientific Data*, *7*(1). https://doi.org/10.1038/s41597-020-00622-y

[4] He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep Residual Learning for Image Recognition. *ArXiv (Cornell University)*. https://doi.org/10.48550/arxiv.1512.03385

[5] Dwivedi, P. (2021, December 7). *Understanding and Coding a ResNet in Keras - Towards Data Science*. Medium. https://towardsdatascience.com/understanding-and-coding-a-resnet-in-keras-446d7ff84d33

[6] Bengio, Y., Simard, P. Y., & Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, *5*(2), 157–166. https://doi.org/10.1109/72.279181

Our GitHub Repository Link: - https://github.com/dp913/CSE-541-Computer-Vision-2023-Group-8