Deep Learning in State-of-the-Art Image Classification Exceeding 99% Accuracy

Chapter · April 2019

DOI: 10.1007/978-3-030-16181-1_89

Emilia Zawadzka-Gosk
Polish-Japanese Academy of Information Technology

Krzysztof Wołk
Polish-Japanese Academy of Information Technology

Deep learning in State-of-The-Art image classification exceeding 99% accuracy.

Emilia Zawadzka-Gosk[1] , Krzysztof Wołk[1], Wojciech Czarnowski[2]

[1] Polish-Japanese Academy of Information Technology, Warsaw, Poland
[2] Jatar, Koszalin, Poland

{ezawadzka, kwolk}@pja.edu.pl, wcz@jatar.com.pl

Abstract. Automatic image recognition and classification is a field of science that became popular in the recent years. Free platforms as Google Collaboratory make machine learning experiments more available to perform for everyone. The current technology enables us to use image recognition in such domains as medicine, criminology, entertainment or trading. In our research we created a state-of-the-art image classifier based on convolutional neural network model to classify ten models of old polish cars. We elaborated eight step training to fine tune the neural network. As the first step the data augmentation and precomputed activations were enabled. After that we froze all the layers but the last one, found the proper learning rate and performed interactive training. Fine-tuning, proper training and appropriate data preparation brought great results. The accuracy of cars' model recognition exceeded 99% with some room for improvement.

Keywords: deep learning, image classification, interactive training, data augmentation, transfer learning.

1 Introduction

Image recognition is a fast-developing part of science [1] [7]. Even few years ago, discovering specified patterns on the pictures was difficult and challenging assignment. Only projects with strong development and scientific teams were able to research this subject. Huge amount of data was also required to perform such experiments. Nowadays, thanks to the development of technology everyone can work on image recognition and classification.

Image recognition becomes an important issue in different domains like medicine [8], security [3], entertainment, trade [14] or science. For example, there are several visual diagnostic methods as MRI [5][6], CT, USG or X-ray [9] [10]. All of them provide for the doctor valuable information about a patient, but not all present patterns are visible to human eyes. The support of automatic image recognition and classification can be a valuable help.

Recognizing human faces [4] can be broadly used in the criminology and security. It can be easily imagined that such technology is utilized by police or security systems. Nowadays it is commonly used for entertainment in smartphones or social media portals, as Facebook.

Object detection is a subject for different everyday activities support, from autonomous cars, assistants for blind or disabled people to detecting systems.

Online trading portals as eBay [14], have the ability to search or verify products available on their website.

Scientists and analysts, working with huge amount of visual data, use classification and recognition for the research.

Recently, the most interesting methods of image recognition are convolutional neural networks. Due to larger availability of big amount of data and great computing power, neural networks are employed more often into tasks of recognition. As the results of CNN usage are highly rewarding, we decided to use convolutional neural network in our work. In our experiment we focused on classifying old polish car brands. The results we got are very satisfying. The ratio of correctly classified images is very high.

2 Environment and dataset

We performed our experiment in Google Collaboratory, the environment provided by Google to experiment with machine learning. It enables to use nVidia K80 graphic cards for computations. It is built on top of Jupyter Notebook, therefore work is convenient and easy to share with other researchers.

Moreover, the PyTorch machine learning library was used in our trainings. We also employed fast.ai, the library to neural network training which facilitates and accelerates the process by including best practices and recent scientific research results [12].

In our work we used dataset containing 10 models of old polish cars:

- Autobus Jelcz MZK
- Autobus Jelcz Ogorek
- Fiat 125p
- Fiat 126p Maluch
- FSC Zuk
- FSO Polonez
- FSO Warszawa
- Gazik
- Syrena 105
- ZSD Nysa.

It consists of 7300 photos of the cars with included information of their brand and 800 photos not signed, prepared for testing. Resolution of the images was reduced to only 224x224px for faster computing.

3 Model and training

For our experiments we used ResNet-34 neural network [2]. It is a convolutional deep network that won the ImageNet 2015 contest. Based on it we prepared a good quality image classifier for old Polish cars' models.

Fine tuning was performed on the ResNet. At first, the last layer of the network was replaced with the new one with 10 outputs (instead of 1000 present in original model). We decided to fit existing model to our data by finding two hyperparameters: learning rate and number of epochs. Hyperparameters describe high-level properties of our model, its complexity and learning pace. They cannot be learnt from the ordinary learning process, but by adjusting the model to the dataset. To perform it we used gradient descent method [15].

To train the network we elaborated eight step training that includes:

1. Enabling the data augmentation and precomputed activations for the training.
2. Finding the highest value of learning rate where the loss value is still decreasing.
3. Freezing all the layers except the last one and training the last layer for 1-2 epochs with precomputed activations.
4. Training the last layer for 2-3 epochs with data augmentation and cycle_len parameter equals to 1.
5. Unfreezing the frozen layers.
6. Setting 3-10 lower learning rates for the previous layers than for the next layers.
7. Finding again the highest value of learning rate where the loss value is still decreasing.
8. Training the whole CNN with cycle_mult parameter equals to 2 until network overfitting.

Figure 1 presents the schema of fine-tuning method performed on our CNN. It needs to be accentuated, that in the first steps, only the last layer is trained, whereas the previous ones are frozen.

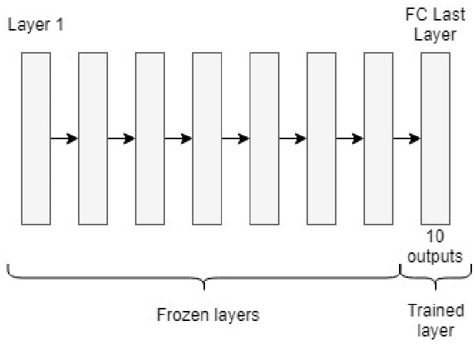

Fig. 1. Schema of CNN during fine-tuning training method

As the first step of our network training we enabled data augmentation and precomputed activations. Next, we found learning rate (speed of our weights updates) using a technique developed in 2015 [13], based on increasing learning rate, starting from very small value and finishing when the loss stopped decreasing.

To validate learning rate of a new (not trained) model, we created new 'learner' and displayed learning rate 'training plan'. Figure 2 displays the learning rate value depending on iterations of SGD algorithm [15].

Additionally, we checked loss value to discover a point where the loss is decreasing (Figure 3). It occurred that the decreasing tendency of the loss value is visible till learning rate value equals 0.1.

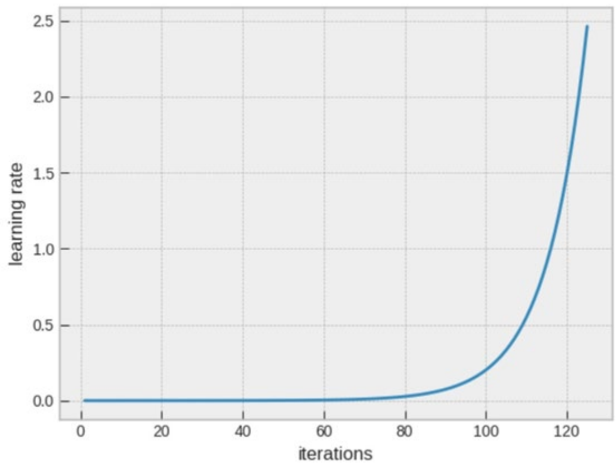

Fig. 2. Learning rate of newly created learner

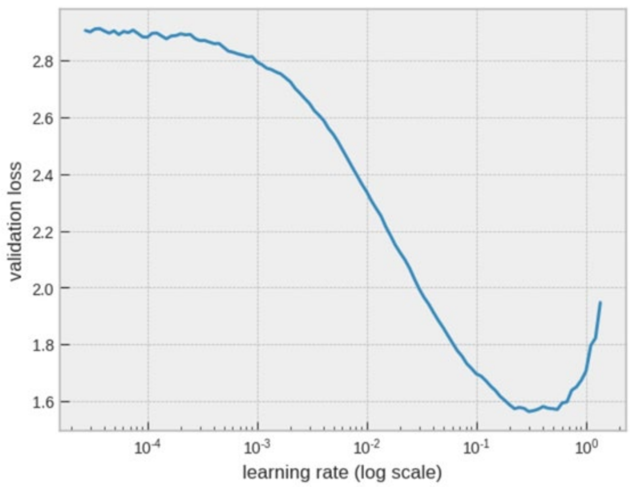

Fig. 3. Loss rate of newly created learner

Analysis of learning rate and loss values made us decide to set the learning rate at 0.08. All the layers of the CNN, except the last one, were frozen.
We trained last layer of the neural network with chosen learning rate for 1 epoch, values of loss and accuracy were as follows: loss=0.24572 and accuracy=0.9173.
Continuing to train the network for more epochs we discovered that the network got over-trained. It means that our model was recognizing individual images from our training set instead of generalizing. It could not be applied to our validation set with have good results. We decided to create more data by augmentation of transforms (Figure 4). The fast.ai library enables to perform augmentation simple and fast, by setting 'aug_tfms' parameter to tfms_from_model. We decided to use transforms_side_on value and max_zoom=1.1, to rotate, flip, change the lighting and enlarge of the pictures.

Fig. 4. Images after augmentation process

After applied data processing we could train the last layer of our CNN by 3 epochs. It resulted with loss=0.20907 and accuracy=0.9295.

We also set cycle_len parameter to 1. In stochastic gradient descent with restarts (learning rate annealing) which we used, it is possible to change the part of the weights' space to find more stable and accurate weights. It is called 'restart'. To force the model to move to another weights' space learning rate is increased rapidly. The cycle_len parameter describes the number of epochs between restarts (Figure 5).

As we already trained the last layer, we would like to adapt the previous layers.

In the model we used previous layers are already trained to recognize images, so we needed to perform carefully to not destroy computed weights. As the previous layers works more generally, they need less tuning then the last ones did. Therefore, we used different learning rates for different layers. First ones: 0.0001, middle layers 0.001 and FC layers 0.01.

Additionally, we used cycle_mult parameter to multiply the length of the learning cycle. The Figure 6 presents the learning rate changes for the last layers where cycle_mult was set to 2.

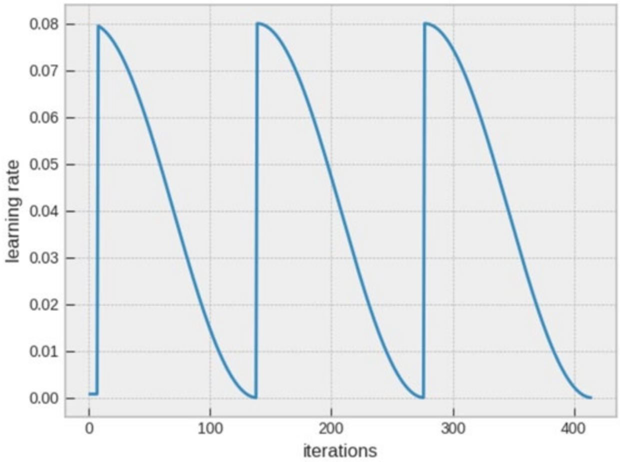

Fig. 5. Learning rate values during training with cycle_len=1

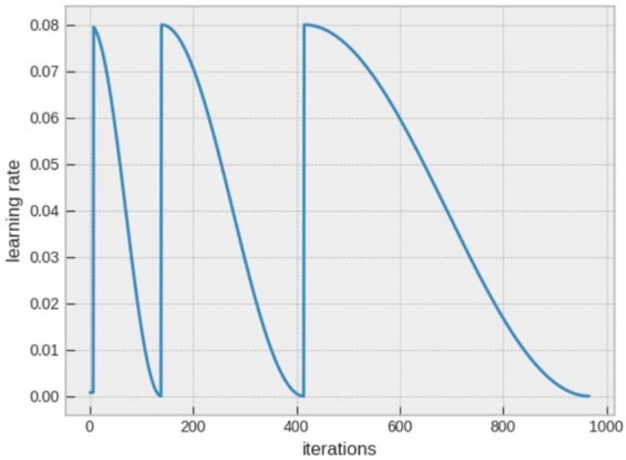

Fig. 6. Learning rate values of the last layers during training with cycle_mult=2.

After those operations we found that our loss value decreased significantly to 0.0456 (Figure 7).

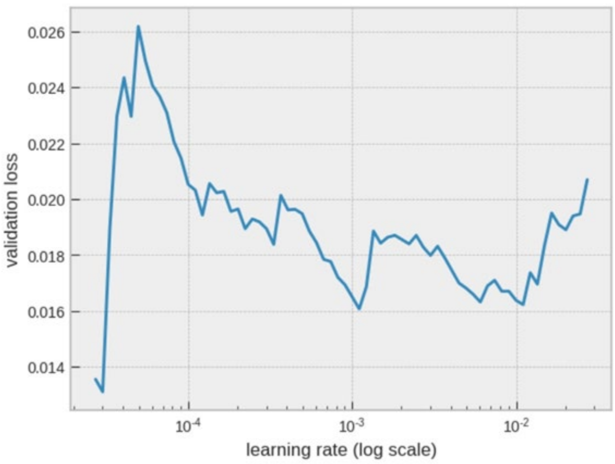

Fig. 7. Loss rate after training previous layers

We continued the training till the network was becoming over-trained. Final accuracy value we received was 0.9918.

To verify our network, we decided to perform Test Time Augmentation [16]. The TTA created predictions based on original image from our dataset and four augmentations of it. The mean prognosis from all those images was also equal to 0.99.

4 Results

The results of our tests without TTA were presented in a confusion matrix (Figure 8). The matrix displays the final classification of old polish cars' models. The results presented on the chart are very good, vast majority of the car images was recognized correctly. Only few images occurred to be categorized wrongly.

In 98 samples of 'Autobus Jelcz MZK' 97 was classified correctly, only one was confused with 'ZSD Nysa'. Similar situation was observed for 'Autobus Jelcz Ogórek', which also was wrongly classified as 'ZSD Nysa'. 'FSO Polonez' was mistaken in one photography with 'FSC Zuk'. 'Fiat 125p' was once classified as 'Fiat 126p Maluch' and the opposite situation also occurred. 'ZSD Nysa' model was also labelled as 'Gazik'. The model with the highest number of classification errors was 'Gazik', 2 of 98 samples were classified incorrectly. Other models were categorized with the accuracy of 100%.

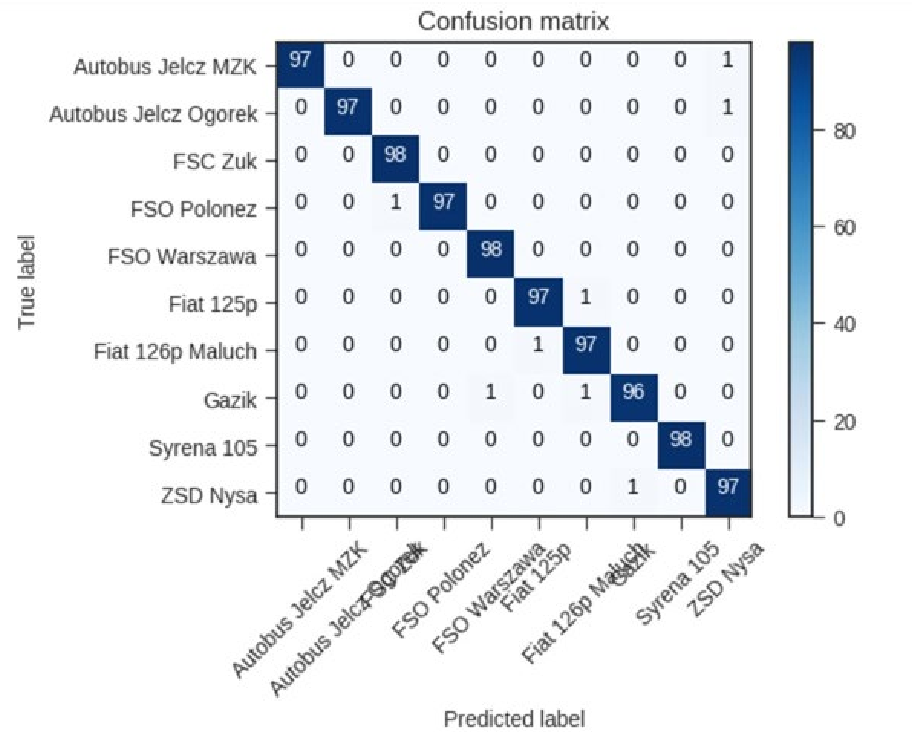

Fig. 8. Confusion matrix of old polish car model's classification system

Having such high value of accuracy, made us more intrigued what images were problematic in classification process. We decided to check what images became recognized correctly and incorrectly in our initial model of CNN, after the initial training of the last layer. We displayed sample pictures of Syrena 105 model which were classified correctly, incorrectly and with uncertain predictions.

Figure 9 presents correctly classified Syrena 105 model. According to the image quality the accuracy of the prediction is diverse. Cars on the first and third images were predicted with high accuracy, over 99%. The vehicles are in the middle of the photography, although the angle of the photoshoot is different. The second image was predicted only for 66,48% to be Syrena 105, but object on the picture is not centred and the photography is probably a part of a larger image.

Images where the photoshoot was made from the front side of the car have the highest probability values, maybe because the hood of this model is very specific. In examples presented in the Figure 10 the system was 100% sure of the model of a car.

Fig. 9. Examples of correctly classified Syrena 105 images

Fig. 10. Examples of correctly classified Syrena 105 images with 100% prediction

The black-white or negatives photos impede the recognition of the car type. Shot at an unusual angle also seems to be problematic for recognition. Examples of such incorrect classifications are present in the Figure 11.

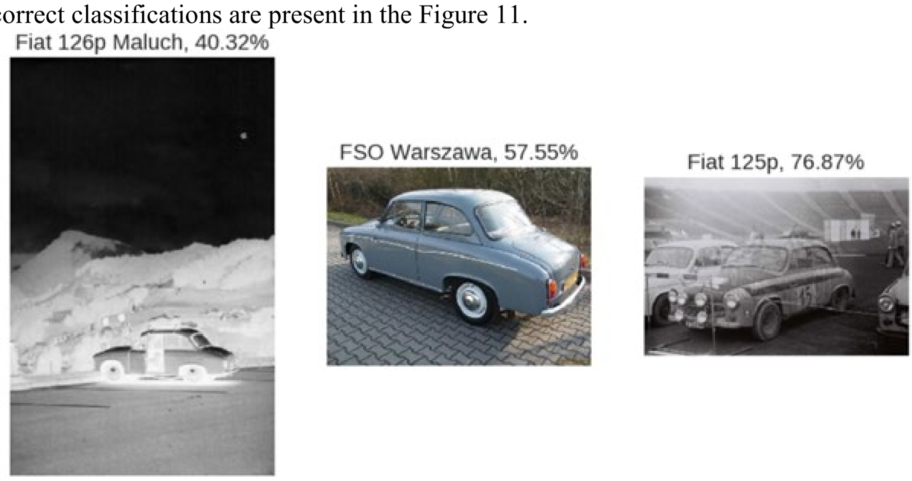

Fig. 11. Examples of incorrectly classified Syrena 105 images

Figure 12 presents most incorrectly categorized cars. All the photos present Syrena 105, but all are classified as other model with quite high probability. Although, it has to be emphasized that classifying car models on these photographs is a challenging task also for human. The photo negative was a very problematic issue to be classified. The CNN classified it as a Fiat 126p with 40% accuracy. Also, the photos where the car was partially covered, or vehicle integrity was modified were categorized incorrectly.

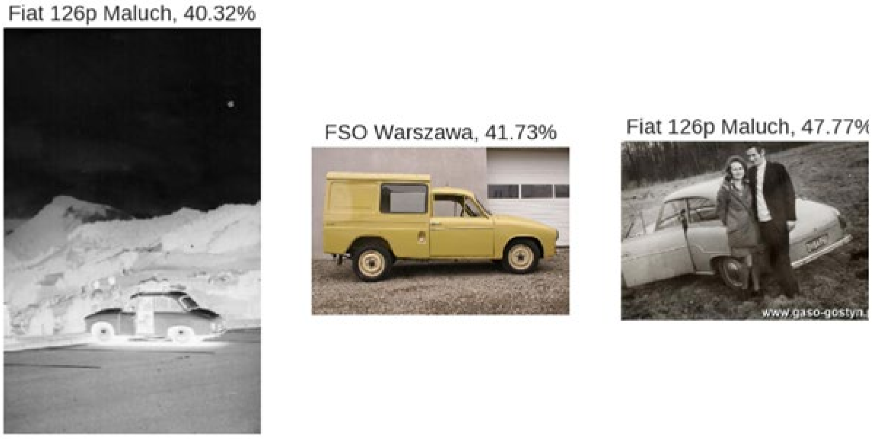

Fig. 12. Examples of most incorrectly classified Syrena 105 images

The example of images that received the most ambiguous results are presented in the Figure 13. All the prediction values are close to 50%. Some of the cars are classified correctly as Syrena 105, but other two as Fiat 126p and Fiat 125p. Again, the unusual angle and the car being just a background of an image caused the problems with model recognition.

Fig. 13. Examples Syrena 105 images uncertain predictions

7 Discussion and conclusion

In this paper we introduced the state-of-the-art image classifier. Using Google Collaboratory, PyTorch, fast.ai and the ResNet-34 neural network model sophisticated and effective image classifier was built. To create it we followed eight steps, including data augmentation, hyperparameters fitting, training of the last and previous neural network layers. We adapted learning rate and cycle parameters using stochastic gradient descent with the restarts method. Providing dataset of over 7000 images of old polish cars the good quality image recognition system was created with accuracy value about 99.18%, which means that the vast majority of the images from validation set was classified correctly. Analysing the individual cases where neural network might have difficulties with classification, we have noticed that the cars on the pictures were difficult to recognize even for humans. Different angles of the photoshoots, partially covered cars or unusual colours of the image seems to be the problem for classification task, but our final solution managed to handle with the issue. After applying our training strategy on the ResNet-34 neural network model we reached more satisfying results, where, according to the confusion matrix, all the pictures of Syrena 105 model were classified correctly.

As our solution still makes incorrect classifications, we plan to continue our research on the issue. Further experiments using ResNet-50 NN model are expected to bring better results, although we will need to use the images of higher resolution like 512x512px or even greater. The accuracy of 99% is very high but still leaves room to improvement. One of the vulnerabilities of our model is poor augmentation. Only basic transformations as rotating, flipping, change of the lighting or picture enlarging were used. Applying more sophisticated graphic editor modifications for training set images can result in better network training. We are thinking about some Photoshop augmentation based on scripts and advanced filters. Finally, we would like to take more closer look at the training parameters of our convolutional neural network and the quality of the training data to improve our final accuracy. Our future work will also expand the scope of our research to the images from other domains, e.g. medical diagnostic images training datasets [11], where high accuracy of implemented solution is a crucial aspect of usefulness.

References

1. Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.
2. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).
3. Daugman, J. (1997). Face and gesture recognition: Overview. IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(7), 675-676.
4. Hong, Z. Q. (1991). Algebraic feature extraction of image for recognition. Pattern recognition, 24(3), 211-219.

5. Hofmann, M., Steinke, F., Scheel, V., Charpiat, G., Farquhar, J., Aschoff, P., ... & Pichler, B. J. (2008). MRI-based attenuation correction for PET/MRI: a novel approach combining pattern recognition and atlas registration. Journal of nuclear medicine, 49(11), 1875.
6. Clarke, L. P., Velthuizen, R. P., Phuphanich, S., Schellenberg, J. D., Arrington, J. A., & Silbiger, M. M. R. I. (1993). MRI: stability of three supervised segmentation techniques. Magnetic Resonance Imaging, 11(1), 95-106.
7. Caputo, B., & Dorko, G. (2003). How to combine color and shape information for 3D object recognition: kernels do the trick. In Advances in Neural information processing systems (pp. 1399-1406).
8. Tadeusiewicz, R., & Ogiela, M. R. (2006). Automatic Image Understanding a New Paradigm for Intelligent Medical Image Analysis. Bio-Algorithms and Med-Systems, 2(3), 3-9.
9. Jain, A. K., & Chen, H. (2004). Matching of dental X-ray images for human identification. Pattern recognition, 37(7), 1519-1532.
10. Kovalev, V., Liauchuk, V., Kalinovsky, A., & Shukelovich, A. (2017). A comparison of conventional and deep learning methods of image classification on a database of chest radiographs. Surgery, 12(1), S139-S140.
11. Kumar, A., Kim, J., Lyndon, D., Fulham, M., & Feng, D. (2017). An ensemble of fine-tuned convolutional neural networks for medical image classification. IEEE journal of biomedical and health informatics, 21(1), 31-40.
12. Howard, J., & Ruder, S. (2018). Fine-tuned Language Models for Text Classification. arXiv preprint arXiv:1801.06146.
13. Smith, L. N. (2017, March). Cyclical learning rates for training neural networks. In Applications of Computer Vision (WACV), 2017 IEEE Winter Conference on (pp. 464-472). IEEE.
14. Find It On eBay: Using Pictures Instead of Words. (2017, July 26). Retrieved November 14, 2018, from https://www.ebayinc.com/stories/news/find-it-on-ebay-using-pictures-instead-of-words/
15. Johnson, R., & Zhang, T. (2013). Accelerating stochastic gradient descent using predictive variance reduction. In Advances in neural information processing systems (pp. 315-323).
16. Now anyone can train Imagenet in 18 minutes. 10 Aug 2018 by Jeremy Howard, from https://www.fast.ai/2018/08/10/fastai-diu-imagenet/