Concept detection on medical images using Deep Residual Learning Network

Katsios Dimitris and Kavallieratou Ergina

Dept. of Information and Communication Systems Engineering, University of the Aegean, Samos, 83200 Greece

Abstract. Medical images are often used in clinical diagnosis. However, interpreting the insights gained from them is often a time-consuming task even for experts. For this reason, there is a need for methods that can automatically approximate the mapping from medical images to condensed textual descriptions. For identifying the presence of relevant biomedical concepts in medical images for the ImageCLEF 2017 Caption concept detection subtask we propose the use of a pretrained residual deep neural network. Specifically, a 50-layered resNet was used and retrained on the medical images. The proposed method achieved F1 Score 0.1583 on the test data.

Keywords: Image retrieval, Concept Detection, Residual Neural Networks, Medical Images.

1 Introduction

Concept detection determines whether an image is relevant to a specific concept. A concept of that type ranges from simple objects (e.g. desk, car) to events (people swimming) or scenes (lecture, sky). Concept detection is considered a challenging task especially in the presence of occlusion, background clutter, intra-class variation, pose and lighting changes in images [1]. Nevertheless, apart from its difficulty it is extremely helpful in tasks like image retrieval where one needs the most relevant images to some concepts from a set of images. While concept detection is not a classification task, it can be solved as one.

Semantic concepts can serve as good intermediate semantic metadata for video content indexing and understanding [2]. Applications of image or video retrieval based on concepts can be met at search engines like google search, social networks like Facebook and content sharing websites like YouTube and Flickr [3]. A prerequisite for effective image and video search is to analyze and index media content automatically. Establishing a large set of robust concept detectors will yield significant improvements in many challenging applications, such as image/video search and summarization [4].

Concept detection is commonly viewed as a supervised machine learning problem which aims to learn the mapping function between low-level visual features and high-level semantic concepts based on the annotated training data [5]. Even if computer vision techniques can solve some of the major problems mentioned above, intra-class variation is one of the most difficult to deal with [6]. Collecting large scale training data to cover a wide variety of samples might be a promising solution since studies on concept detections [7] and pedestrian classification [8, 9] indicate that data matters most, since the amount of available data impacts to the accuracy of a classification model more than other parameters.

Apart from present applications that are based on concept detection, there are many future goals that could be matched by this technique. One goal would be the automatic clinical diagnosis based on patient images. To achieve it one should be able to detect which medical concepts are present at each image and then combine this knowledge to extract a patient status description in natural language. This would be very helpful since diagnosis is a time-consuming task even for highly trained experts.

One of the most popular approaches in this domain is the Bag of Words (BoW) which transforms local image descriptors into image representations [10, 11, 12]. BoW image representation is analogous to the BoW representation for text documents which means that techniques from the second can be applied to semantic concept detection. These models extract local descriptors from images, embed them to a visual vocabulary space and compute statistics based on the occurrences of each visual word in the image while some models use co-occurrences of (visual) words or other higher-order occurrence pooling. Some standard BoW methods are Local Coordinate Coding [13], Sparse Coding [14, 15], Approximate Locality-constrained Linear Coding [16], Approximate Locality-constrained Soft Assignment [17, 18] and Soft Assignment and Visual Word Uncertainty [19, 20, 21]. There is another group of approaches which use more advanced techniques like Super Vector Coding [22], Fisher Vector Encoding [23, 24], Vector of Locally Aggregated Descriptors [25], and Vector of Locally Aggregated Tensors [26]. One important feature in BoW is the representation choice.

Some representations are related to text categorization techniques, like stop word removal, word weighting scheme, visual bigram and feature selection, while the others are unique to concept detection in videos or images, like spatial information of the key points or vocabulary size (number of keypoint clusters). Research is mainly focused on finding better keypoint descriptors, keypoint detectors and clustering algorithms [27, 28] or what representation choices (weighting, selection, w.r.t dimension) have better accuracy and efficiency. Some methods [29, 30] use different keypoint sampling methods including sparse detectors as Boosted ColorHarris Laplace as well as keypoint descriptors like SIFT and HueSIFT. Geometric blur features is another way some methods used local features as keypoint descriptor for concept detection [31]. Dense and sparse representation with grid-based local image patches at the first category and keypoints at the second were compared [32] based on different sampling strategies of BoW with the results indicating that randomly sampled image patches offer better representation characteristics.

Apart from BoW new methods have been developed to improve concept detection accuracy like Deep Convolutional Neural Networks [33]. This has been mainly achieved due to the available pool of features which in recent years has increased rapidly. Deep Convolutional Neural Networks (CNNs) can be combined with other visual descriptors to improve its performance [34], which is overall better than previous approaches [35]. One of the most recently developed architectures of Deep Neural Networks (NNs) is the Deep Residual Learning Network. Deep Residual NN is a network that was developed by researchers from Microsoft Research which received first place in ImageNet Large Scale Visual Recognition Competition (ILSVRC) 2015 image classification. The network that they used had 152 layers, 8 times deeper than a comparable Visual Geometry Group (VGG) network but still having lower complexity. This was the first of many recent applications of Residual Networks. However, this type of networks has not yet been widely implemented at the image concept detection domain. Microsoft Research team developed a variety of Residual Networks, one of which was used in our approach. Specifically, we used the 50-layered network that they developed with some modifications. Due to the nature of the ImageCLEF 2017 Concept Detection subtask, the network must be implemented on multilabeled data, so that for each medical image, more than one label can be assigned. This differs from the usual classification problem where each image belongs to only one class. For this reason, the network output must be not a scalar but a 20,464-long vector, one for each one of the 20,464 potential labels.

In section 2 the proposed technique is presented in detail, while in section 3 experimental and comparison results are given. Finally, in section 4, our conclusion is drawn and some ideas for future work.

2 Proposed System

Training images included a very large variety of situations, content and types, from radiology X-rays and clinical photographs to charts and equipment images. Fig. 1 includes some images from training and validation datasets where one can observe the wide variety. For this reason, context specific descriptors might be difficult to be defined. A neural network with large enough depth might be more suitable for managing this variety and for that reason it was selected as the main method. Residual neural networks are networks that due to its layer modules and connectivity can go deeper while avoiding the degradation problem. Fig. 2 shows the building block of such a network. Residual neural networks were used in the past for image classification with very promising results. Also, according to many researchers like [36, 37] the use of pretrained networks even from a quite different topic is better as starting point. Pretrained networks are networks that have been trained on another task and then the layers with their weights can be used as starting point for a new task.

To train a Deep NN, one can use a framework like Caffe, Torch, Theano, TensorFlow etc. which provides utilities in terms of develop, change, tune, train and test networks of different architectures. Each framework supports specific data types not only for the network description but for the training and test data as well. Since knowledge transfer is one of the most useful concepts in machine learning generally and in deep NNs specifically, networks that were not only suitable in terms of architecture but also pretrained on a similar task were searched. Residual Neural Networks have been proved to be able to achieve much better performance in general image classification [38]. For this reason, it was one of the network types that was tried for this task.

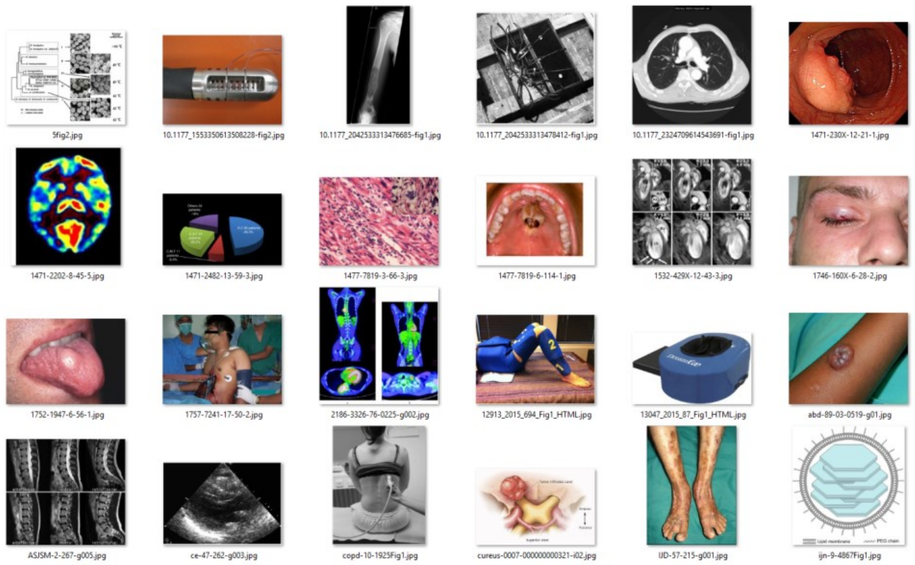

Fig. 1. Medical images sample: main characteristic of training set is the great variety. Training set images of ImageCLEF caption 2017 (Eickhoff et al. 2017)

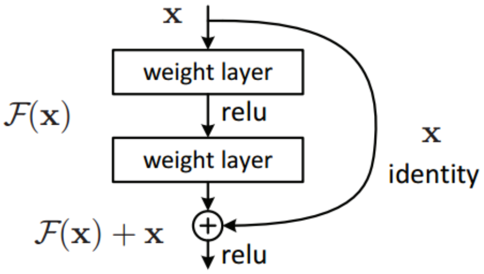

Fig. 2. Residual learning: a building block (He, Kaiming, et al. "Deep residual learning for image recognition, 2016)

Caffe Model Zoo is a framework supported by Berkeley Vision and Learning Center (BVLC) that hosts in GithubGist format different pretrained models for other researchers to download and use. Some of these networks were used and retrained for the concept detection subtask, namely Pascal VOC 2012 Multilabel Classification Model [39], Residual Networks Models by CVGJ (10 layers) [40], ResNet-50 and ResNet-101. Also, an attempt was made to develop and train some networks from scratch. The one with the best performance was ResNet-50 [41] as described in [38]. This network was developed by Microsoft Research Asia (MSRA) and trained on ImageNet and COCO datasets (2015) for the corresponding competitions. Both training sets had images and labels of "general content" meaning not targeted to a specific domain, in contrast to the ImageCLEF caption task which included medical-oriented images and labels.

The three basic components of a network are its architecture, the weights and the parameters. At Caffe framework [42], there is a clear separation between these three even at data level. For each layer of the network that was not modified, the initial weights were kept as is, since knowledge transfer was the main purpose of using pre-trained networks. The training parameters were tuned for optimal performance and best accuracy of the network. As for the net architecture, some modifications were necessary. The original network was trained on simple labeled images, which means that to each (part of) image corresponded (at most) one label. In contrast, for each medical image of ImageCLEF caption, more than one label could be matched, which means that it is a multilabeled task. Because of this difference, the file format of the inputs of the network had to change. In most cases the network inputs for the training phase consist of the image and the corresponding label. However, in our case the input should be the medical image and a list of zero or more labels. Caffe framework supports this type of inputs based on a specific file format, named HDF5. HDF5 allows us to handle the existence of more than one labels for each training image. The data file transformation took place at batches of 500 images with the standard python library h5py. As result 330 such files were created and used as input files for the network.

This transformation led to changes at the first (input) layer of the net. Another change that had to be done was at the last two layers, the fully connected penultimate layer and the loss layer. Specifically, the fully connected layer should have 20,464 outputs, one for each potential label. These potential labels are the ones extracted from the training set. These outputs are the inputs of the last layer, the loss function layer. In most of the cases a SoftMax layer is used to give the possibility of each label to match the specific image and the one with the maximum likelihood is selected for the loss computation. However, in our case the loss function should be able to handle more than one labels. For this another change took place at the original network. Summarizing, the changes over the original ResNet-50 [41] network are:

- the type of the first layer changed to HDF5Data with batch size 2 (see Fig. 3)
- the number of outputs of the last inner product layer changed to 20.464 which is the number of different labels as extracted from the training data (see Fig. 4)
- a new SigmoidCrossEntropyLoss layer was added as the final layer of the network with bottom layers the aforementioned inner product layer and the label layer. This change took place due to the necessity of computing the loss function for multiple outputs each time and not just for the best fitted label (see Fig. 4).

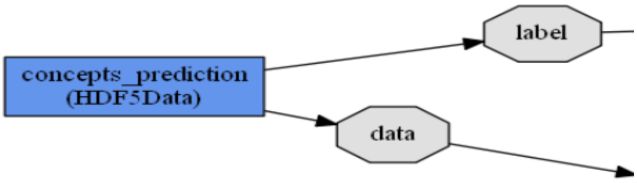

Fig. 3. First layer (HDF5Data)

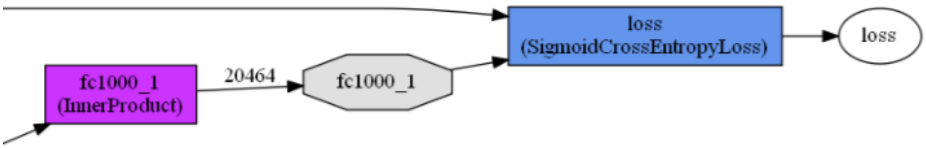

Fig. 4. Last layers (InnerProduct-20464, SigmoidCrossEntropyLoss)

The final network architecture is shown in Fig. 5.

After the network configuration, the training phase took place. In order to optimize the network both in terms of time/resources consumption and results accuracy, proper hyper-parameters had to be set. Some well-known ranges for parameterizing deep nets are momentum ~ 0,9 and weight decay ~ 0,0005 [43]. Of course, the most important set of parameters has to do with the learning procedure. Specifically, one can select from predefined learning policies like fixed, inv, step, multistep, poly etc. Each learning rate policy decreases the learning rate as the learning process progresses in a different way. For example, step policy returns:

$$lr = base\_lr \cdot gamma^{\left(floor\left(\frac{iter}{step}\right)\right)} \tag{1}$$

where *lr* is the learning rate, *base_lr* is the starting point for learning rate, *gamma* and *step* are hyperparameters and *iter* is the iteration of the training procedure.

This means that every *step* iterations, learning rate will be decreased by a factor *gamma*. In case of *gamma* = 0,1 (a common *gamma* value) this means that learning rate will be divided by 10 every *step* iterations. In our case the learning policy of the solver was step with *gamma* 0,1 and base learning rate 10e-6. The base learning rate had to be small enough for the system to be able to compute the loss at each iteration since higher learning rates could not converge. The step size was set to 25,000 iterations so that enough epochs could pass at each step. Another parameter that had to be defined is the iteration size. Iteration size works together with the batch size defined at the inputs layer of the network so that the actual batch of the back propagation is the product of these values. This means that with batch size 2 and iteration size 50, the back propagation with batch Stochastic Gradient Descent will compute the gradient against 100 training examples.

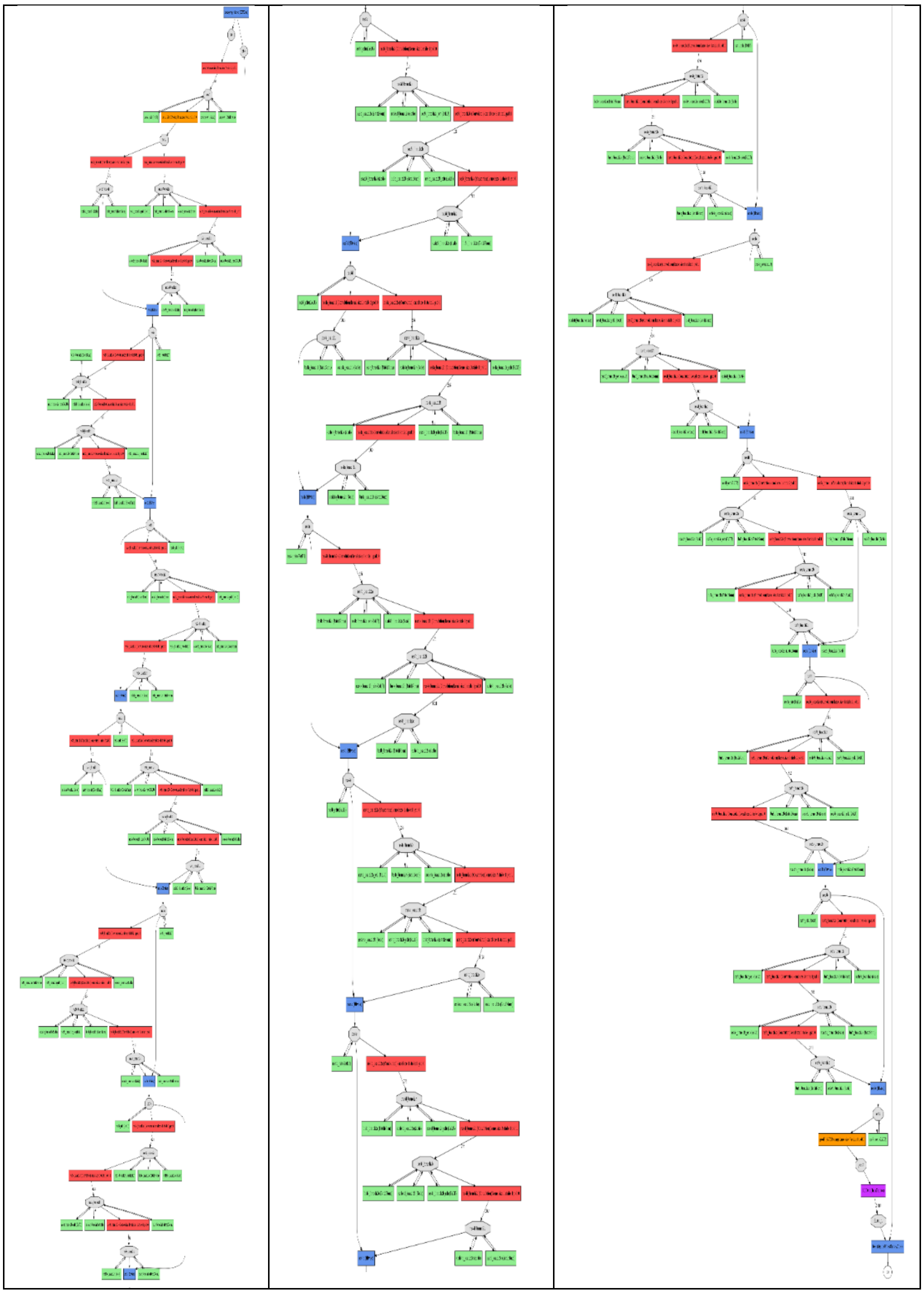

Fig. 5. Proposed Residual Network architecture. Due to the size of the network representation, the figure was split in three parts for clarity, starting from left to right. Red rectangles are the convolution layers, blue rectangles are the elementwise operators and green rectangles are other types of layers like Scale, BatchNorm, ReLU etc. Caffe annotation has been used.

The benefit of this combination is that the maximum value for batch size based on the memory limitations of a very deep network (2) was used and the phenomenal batch size was increased to 100 with the iteration size which accelerates the back-propagation procedure. Summarizing, the network training parameters were:

- learning rate: 10e-6
- learning policy: step
- gamma: 0.1
- stepsize: 25,000
- momentum: 0.9
- weight decay: 0.0005
- iteration size: 50

One important informal parameter that had to be tuned is the threshold of label acceptance. Each one of the 20,464 outputs of the network, is a real number which determines if the corresponding label is predicted to be relative to the image or not. This real number must be transformed to boolean, to procced with the accuracy computations. Normally a threshold equal to 0, 0.5 or 1 is selected so that negative values denote rejection of the label and positive values acceptance (in case of 0) or values close to 0 for rejection and close to 1 for acceptance (in case of 0.5) etc. In our case, the tuning procedure showed that the best accuracy levels were obtained with a threshold of -2. For this network and these parameters, -2 was almost every time the best choice. However, this value was not the best choice when training either other networks, or the same network (ResNet-50) with other hyperparameters. In these cases, the best suited threshold ranged from -4 to 3.

3 Experimental data and results

In order to train the network, the official training dataset of ImageCLEF caption category [44] of ImageCLEF 2017 [45] was used. This dataset contains 164,614 biomedical images extracted from scholarly articles on PubMed Central. Since the given images had no fixed size a data preparation process was necessary. The preparation of data includes the Histogram Equalization for all three channels (RGB) with the standard OpenCV library (cv2.equalizeHist) and image resize to 227 x 227 pixels. Both steps are standard for neural network inputs preparation since the input layer of the proposed network has fixed size (227x227x3) and must be normalized. The same holds for validation and test images as well.

Many attempts took place in order to find the appropriate values for each hyperparameter. For the tuning of threshold of label acceptance, for example several thresholds were tried until the value -2 was selected. In Fig. 6 the difference in accuracy is shown when changing the threshold value for a specific iteration. As one could expect the curve of the relation between threshold and accuracy is concave with a total maximum at some point. In the case of this network all the experiments showed -2 as optimal value.

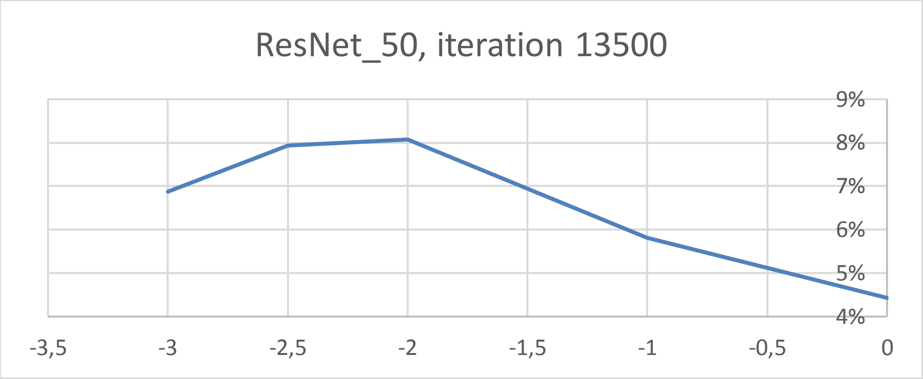

Fig. 6. Threshold of label acceptance (iteration 13500)

Another training parameter, that was tuned, is the momentum term. As mentioned before a value of 0.9 was selected, however in different learning phases, other momentum values had better performance. For example, in Fig. 7 one can see the accuracy levels for iterations 250,000 to 350,000 for two different momentum values: 0.7 and 0.99.

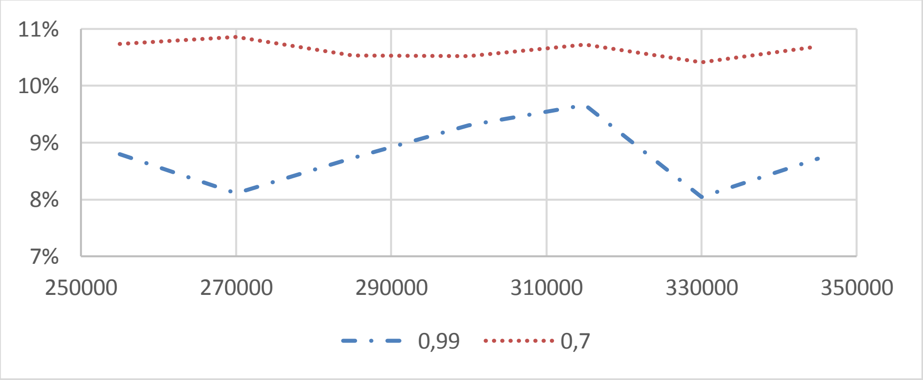

Fig. 7. Accuracy levels for iterations 250,000 to 350,000 for momentum values 0.7 and 0.99

In Fig. 8 one can see the training curve of the network. The accuracy percentage pick happens at 32,000 iterations. This is the model (weights) of the proposed network that was selected for the test run. The accuracy level over the validation data was 11.399%, while the F1 score of the same model (DET_ConceptDetectionTesting2017-results.txt) on the test data of ImageCLEF caption was 0.1583.

It is important to mention the steep increase in accuracy levels that happens at 25,000 iterations, when the learning rate changes from 10e-6 to 10e-7. However, short after this improvement the network performance converged to this level.

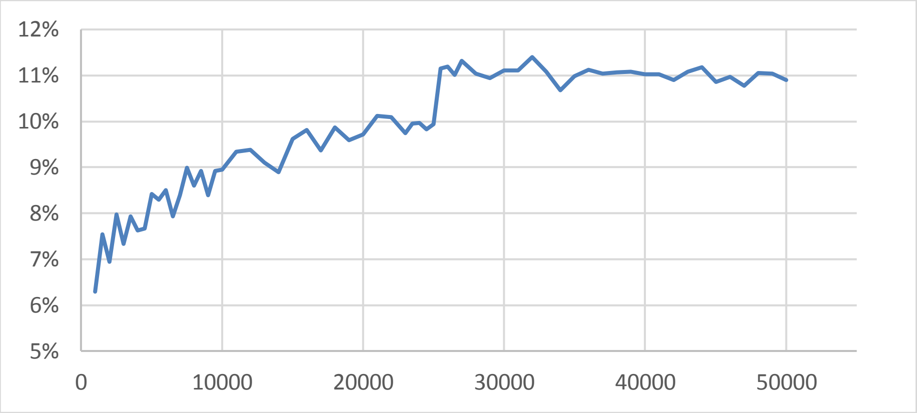

Fig. 8. Training curve of network. Steep increase at 25,000 iterations and peak at 32,000 iterations

The same behavior can be observed at other learning rate changes too. For example, in Fig. 9 the same sharp improvement happens when passing from learning rate 10e-6 to 10e-7 for a different iteration size.

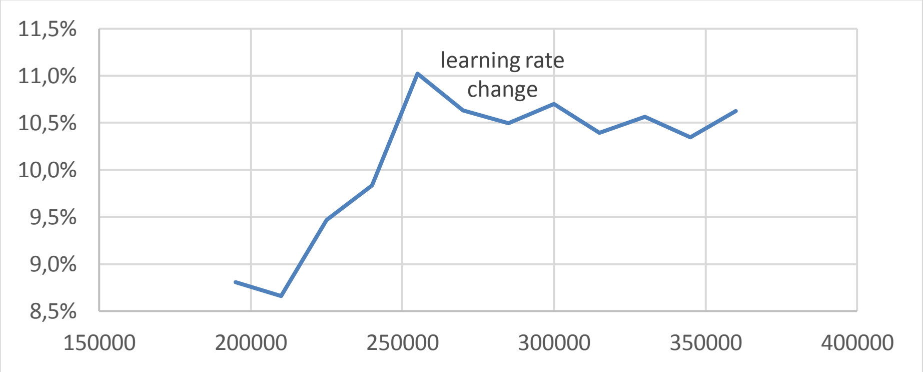

Fig. 9. Accuracy improvement with learning rate change from 10e-6 to 10e-7

4 Conclusion – Future work

The Concept Detection subtask of ImageCLEF caption 2017 required the identification of the presence of relevant biomedical concepts in the medical images. A training dataset included 164,614 biomedical images extracted from scholarly articles on PubMed Central was proposed. The difficulty of this task emerges from three different key points. Firstly, the images variety and diversity was very large as one can see on the image examples (see Fig. 1) which limits the use of content specific descriptors. Secondly, the range of labels that could be assigned to each medical image was very large i.e. more than 20,000 labels, when many labeling and classification tasks even today handle a few hundreds of labels. The number of labels of the specific subtask was extremely big for present methods and resources to handle. The third and final key point of difficulty was the fact that the specific subtask was about multilabeled data. This means that each image does not belong to one and only class so that one label has to be assigned to it, but in contrast each image could belong to more than one classes meaning more than one labels could be assigned to it.

Deep Residual networks proved able to handle better the diversity and complexity of the medical images than other types of shallower networks. Also, pretrained networks had better performance than training them from scratch while retaining the same architecture and parameters.

For the future work, we seek further improvements by applying clustering of the images and using different networks for training over each cluster or ensemble methods together with deep neural networks for better performance. Clustering was attempted over the training data, however when 5±1 clusters were constructed with different clustering algorithms, more than 90% of the training images were assigned to one cluster, which disused the approach’s philosophy. Another promising approach could be the use of different thresholds of acceptance for each label to optimize the detection procedure. This means that instead of having one threshold value for all the labels (e.g. -2), one could use a vector where each value would be the threshold of acceptance for each label. This would mean that the optimal value for each label should be obtained for each network model and then the one with maximum accuracy could be selected.

References

1. Jiang, Yu-Gang, et al. "Representations of keypoint-based semantic concept detection: A comprehensive study." IEEE Transactions on Multimedia 12.1 (2010): 42-53.
2. Tang, Sheng, et al. "Sparse ensemble learning for concept detection." IEEE Transactions on Multimedia 14.1 (2012): 43-54.
3. Huiskes, Mark J., Bart Thomee, and Michael S. Lew. "New trends and ideas in visual concept detection: the MIR flickr retrieval evaluation initiative." Proceedings of the international conference on Multimedia information retrieval. ACM, 2010.
4. Zhu, Shiai, et al. "On the sampling of web images for learning visual concept classifiers." Proceedings of the ACM International Conference on Image and Video Retrieval. ACM, 2010.
5. Bao, Lei, et al. "Boosted Near-miss Under-sampling on SVM ensembles for concept detection in large-scale imbalanced datasets." Neurocomputing 172 (2016): 198-206.
6. Sun, Yongqing, Kyoko Sudo, and Yukinobu Taniguchi. "Visual concept detection of web images based on group sparse ensemble learning." Multimedia Tools and Applications 75.3 (2016): 1409-1425.
7. Huiskes, Mark J., Bart Thomee, and Michael S. Lew. "New trends and ideas in visual concept detection: the MIR flickr retrieval evaluation initiative." Proceedings of the international conference on Multimedia information retrieval. ACM, 2010.
8. Enzweiler, Markus, and Dariu M. Gavrila. "Monocular pedestrian detection: Survey and experiments." IEEE transactions on pattern analysis and machine intelligence 31.12 (2009): 2179-2195.

9. Munder, Stefan, and Dariu M. Gavrila. "An experimental study on pedestrian classification." IEEE transactions on pattern analysis and machine intelligence 28.11 (2006): 1863-1868.
10. Lowe, David G. "Object recognition from local scale-invariant features." Computer vision, 1999. The proceedings of the seventh IEEE international conference on. Vol. 2. Ieee, 1999.
11. Mikolajczyk, Krystian, and Cordelia Schmid. "A performance evaluation of local descriptors." IEEE transactions on pattern analysis and machine intelligence 27.10 (2005): 1615-1630.
12. Sande, K. E. A., T. Gevers, and C. G. M. Snoek. "A comparison of color features for visual concept classification." (2008): 141-149.
13. Yu, Kai, Tong Zhang, and Yihong Gong. "Nonlinear learning using local coordinate coding." Advances in neural information processing systems. 2009.
14. Lee, Honglak, et al. "Efficient sparse coding algorithms." Advances in neural information processing systems 19 (2007): 801.
15. Yang, Jianchao, et al. "Linear spatial pyramid matching using sparse coding for image classification." Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. IEEE, 2009.
16. Wang, Jinjun, et al. "Locality-constrained linear coding for image classification." Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on. IEEE, 2010.
17. Liu, Lingqiao, Lei Wang, and Xinwang Liu. "In defense of soft-assignment coding." Computer Vision (ICCV), 2011 IEEE International Conference on. IEEE, 2011.
18. Koniusz, Piotr, Fei Yan, and Krystian Mikolajczyk. "Comparison of mid-level feature coding approaches and pooling strategies in visual concept detection." Computer vision and image understanding 117.5 (2013): 479-492.
19. Van Gemert, Jan C., et al. "Visual word ambiguity." IEEE transactions on pattern analysis and machine intelligence 32.7 (2010): 1271-1283.
20. Philbin, James, et al. "Lost in quantization: Improving particular object retrieval in large scale image databases." Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on. IEEE, 2008.
21. Koniusz, Piotr, and Krystian Mikolajczyk. "Soft assignment of visual words as linear coordinate coding and optimisation of its reconstruction error." Image Processing (ICIP), 2011 18th IEEE International Conference on. IEEE, 2011.
22. Zhou, Xi, et al. "Image classification using super-vector coding of local image descriptors." Computer Vision–ECCV 2010 (2010): 141-154.
23. Perronnin, Florent, and Christopher Dance. "Fisher kernels on visual vocabularies for image categorization." Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on. IEEE, 2007.
24. Perronnin, Florent, Jorge Sánchez, and Thomas Mensink. "Improving the fisher kernel for large-scale image classification." Computer Vision–ECCV 2010 (2010): 143-156.
25. Jégou, Hervé, et al. "Aggregating local descriptors into a compact image representation." Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on. IEEE, 2010.
26. Negrel, Romain, David Picard, and Philippe-Henri Gosselin. "Compact tensor based image representation for similarity search." Image Processing (ICIP), 2012 19th IEEE International Conference on. IEEE, 2012.
27. Sivic, Josef, and Andrew Zisserman. "Video google: A text retrieval approach to object matching in videos." iccv. Vol. 2. No. 1470. 2003.
28. Zhang, Jianguo, et al. Local features and kernels for classification of texture and object categories: An in-depth study. Diss. INRIA, 2005.
29. Snoek, Cees, et al. "The MediaMill TRECVID 2009 semantic video search engine." TRECVID workshop. 2009.

30. Sande, K. E. A., T. Gevers, and C. G. M. Snoek. "A comparison of color features for visual concept classification." (2008): 141-149.
31. Berg, Alexander C., and Jitendra Malik. "Geometric blur for template matching." Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on. Vol. 1. IEEE, 2001.
32. Nowak, Eric, Frédéric Jurie, and Bill Triggs. "Sampling strategies for bag-of-features image classification." Computer Vision–ECCV 2006 (2006): 490-503.
33. Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." Advances in neural information processing systems. 2012.
34. Markatopoulou, Foteini, Vasileios Mezaris, and Ioannis Patras. "Ordering of visual descriptors in a classifier cascade towards improved video concept detection." International Conference on Multimedia Modeling. Springer International Publishing, 2016.
35. Koniusz, Piotr, et al. "Higher-order occurrence pooling for bags-of-words: Visual concept detection." IEEE transactions on pattern analysis and machine intelligence 39.2 (2017): 313-326.
36. Yosinski, Jason, et al. "How transferable are features in deep neural networks?." Advances in neural information processing systems. 2014.
37. Sharif Razavian, Ali, et al. "CNN features off-the-shelf: an astounding baseline for recognition." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. 2014.
38. He, Kaiming, et al. "Deep residual learning for image recognition." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016.
39. Lapuschkin, Sebastian, et al. "Analyzing classifiers: Fisher vectors and deep neural networks." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016.
40. Simon, Marcel, Erik Rodner, and Joachim Denzler. "ImageNet pre-trained models with batch normalization." arXiv preprint arXiv:1612.01452 (2016).
41. Deep Residual Learning for Image Recognition GitHub https://github.com/KaimingHe/deep-residual-networks
42. Jia, Yangqing, et al. "Caffe: Convolutional architecture for fast feature embedding." Proceedings of the 22nd ACM international conference on Multimedia. ACM, 2014.
43. Zhou, Aojun, et al. "Incremental network quantization: Towards lossless cnns with low-precision weights." arXiv preprint arXiv:1702.03044 (2017).
44. Carsten Eickhoff, Immanuel Schwall, Alba García Seco de Herrera and Henning Müller. Overview of ImageCLEFcaption 2017 - Image Caption Prediction and Concept Extraction Tasks to Understand Biomedical Images, CLEF Labs Working Notes, CEUR, 2017.
45. Bogdan Ionescu, Henning Müller, Mauricio Villegas, Helbert Arenas, Giulia Boato, Duc-Tien Dang-Nguyen, Yashin Dicente Cid, Carsten Eickhoff, Alba Garcia Seco de Herrera, Cathal Gurrin, Bayzidul Islam, Vassili Kovalev, Vitali Liauchuk, Josiane Mothe, Luca Piras, Michael Riegler, Immanuel Schwall, Overview of ImageCLEF 2017: Information extraction from images, CLEF 2017 Proceedings, Springer LNCS, 2017.