Cutting the Error by Half: Investigation of Very Deep CNN and Advanced Training Strategies for Document Image Classification

4 authors:

Muhammad Zeshan Afzal
Technische Universität Kaiserslautern
43 PUBLICATIONS **254** CITATIONS

SEE PROFILE

Andreas Kölsch
Technische Universität Kaiserslautern
4 PUBLICATIONS **14** CITATIONS

SEE PROFILE

Sheraz Ahmed
Deutsches Forschungszentrum für Künstliche Intelligenz
51 PUBLICATIONS **301** CITATIONS

SEE PROFILE

Marcus Liwicki
Université de Fribourg
244 PUBLICATIONS **3,335** CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Project — Subword Semantic Hashing for Intent Classification on Small Datasets View project

Project — metis – Knowledge-based search and query methods for the development of semantic information models (BIM) for use in early design phases View project

Cutting the Error by Half: Investigation of Very Deep CNN and Advanced Training Strategies for Document Image Classification

Muhammad Zeshan Afzal*†§, Andreas Kölsch*†, Sheraz Ahmed‡, Marcus Liwicki†¶

afzal@iupr.com, a_koelsch12@cs.uni-kl.de, sheraz.ahmed@dfki.de, marcus.liwicki@unifr.ch

†MindGarage, University of Kaiserslautern, Germany
‡DFKI, Kaiserslautern, Germany
§Insiders Technologies GmbH, Kaiserslautern, Germany
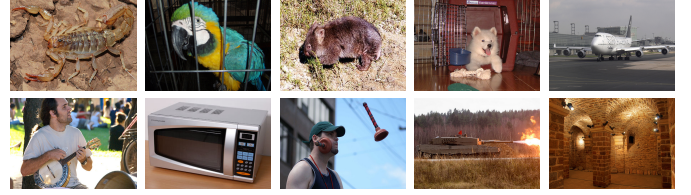¶University of Fribourg, Switzerland

Abstract—We present an exhaustive investigation of recent Deep Learning architectures, algorithms, and strategies for the task of document image classification to finally reduce the error by more than half. Existing approaches, such as the DeepDoc-Classifier, apply standard Convolutional Network architectures with transfer learning from the object recognition domain. The contribution of the paper is threefold: First, it investigates recently introduced very deep neural network architectures (GoogLeNet, VGG, ResNet) using transfer learning (from real images). Second, it proposes transfer learning from a huge set of document images, i.e. $400,000$ documents. Third, it analyzes the impact of the amount of training data (document images) and other parameters to the classification abilities. We use two datasets, the Tobacco-3482 and the large-scale RVL-CDIP dataset. We achieve an accuracy of 91.13% for the Tobacco-3482 dataset while earlier approaches reach only 77.6%. Thus, a relative error reduction of more than 60% is achieved. For the large dataset RVL-CDIP, an accuracy of 90.97% is achieved, corresponding to a relative error reduction of 11.5%.

Index Terms—Document Image Classification, Deep CNN, Convolutional Neural Network, Transfer Learning

(a) Sample images from the ImageNet dataset

(b) Sample images from the RVL-CDIP dataset

Fig. 1: Sample Images from Imagenet and RVL-CDIP datasets are shown in (a) and (b) respectively.

I. INTRODUCTION

An important step of the Document Image Processing Pipeline (DIPP) is the classification of the documents. An early classification of documents helps to process the subsequent processes in DIPP such as information extraction, text recognition etc [1]. Due to its fundamental importance, this area has been explored extensively. Earlier methods that have been dealing with document classification focused mainly on either exploiting the structural similarity constraints [2], [3] or extracting features from the documents that may be able to help for document classification [4]–[6]. Some of the methods combine both of the features [7].

Deep Learning has been used for many document analysis tasks such as binarization [8], [9], layout analysis [10], [11], Optical Character Recognition (OCR) [12]–[17] etc. Recently, deep learning methods have also been exploited for document image classification [18]–[20]. Deep learning methods do not require any manual feature extraction. However, the existing state-of-the-art methods do transfer learning. Fig. 1 shows the sample images both real and document images from the ImageNet [21] and Tobacco-3482 datasets respectively. While the images are visually very different, the visual queues are generic and thus, transfer learning helps to boost the performance of the document image classification [18], [20]. The networks that are not using transfer learning (i.e., they are randomly initialized) are under-performing [19]. The performance evaluation for the deep neural networks was only performed using Tobacco-3482 images. Another dataset introduced by Harley et al. [20] consist of $400,0000$ images that are divided into 16 classes. Representative images from each of the classes are shown in Fig. 2. Although now we have a large dataset available for training document images, there is no study that shows the performance of very deep networks for large datasets of document images. Furthermore, the potential of pretraining using document images only is not explored either.

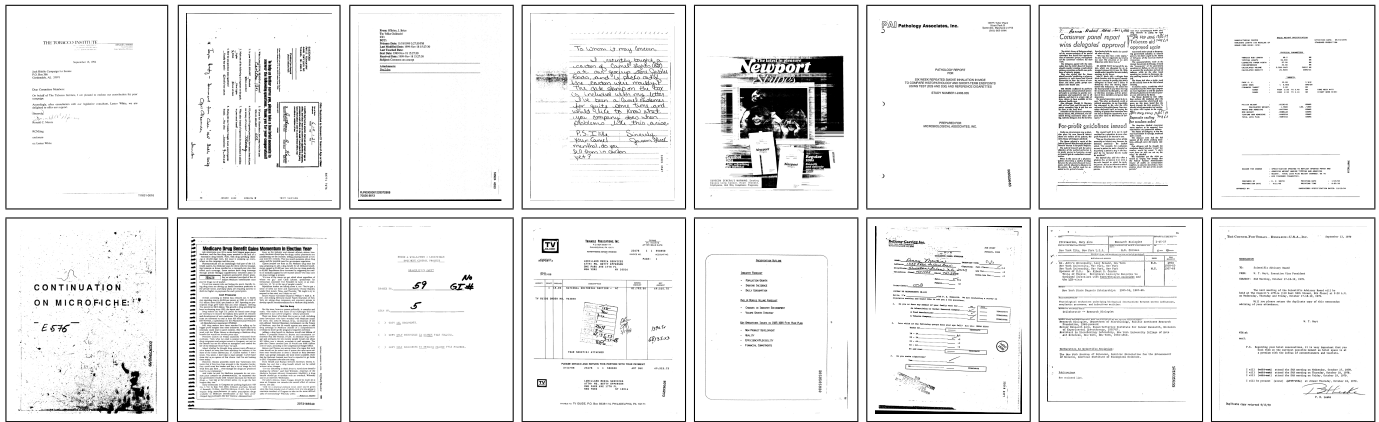* These authors contributed equally to this work

Fig. 2: Sample images from the RVL-CDIP dataset. One image from each class is depicted. From left to right: *Letter*, *Form*, *Email*, *Handwritten*, *Advertisement*, *Scientific report*, *Scientific publication*, *Specification*, *File folder*, *News article*, *Budget*, *Invoice*, *Presentation*, *Questionnaire*, *Resume*, *Memo*

Therefore, in this work, we evaluate deep neural networks both on the small and big dataset. An exhaustive evaluation of the deep neural networks is performed to show the impact of the amount of data for training in combination with very deep Convolutional Neural Network (CNN). Furthermore, an evaluation is performed for transfer learning when the weights are adapted both from natural images and document images. The proposed approach shows a significant improvement over the current state-of-the-art by reducing the error by more than half.

II. RELATED WORK

Over the years, different methods have been proposed for document image classification. The overall classification methods can be divided into three distinct categories. The first category exploits structure and layout similarities, while the second focuses on developing local and global features that could be used for document classification. The third category is based on deep CNNs that extract the features automatically for document classification. This section provides a summary of the important related work regarding the above mentioned three categories.

Dengel and Dubiel [1] used layout structure printed documents. They used top-down induction in decision trees to convert printed documents into a complementary logical structure. Bagdanov and Worring [22] classify machine-printed documents by using the Attributed Relational Graphs (ARGs). Byun and Lee [2] used parts of the documents for the recognition. They reasoned that processing complete documents is time-consuming. The document classification was performed on parts of the documents using DP algorithm. Their approach was fast but the applicability is limited to forms. Shin and Doermann [3] proposed an approach that used layout structural similarity for full or partial image matching for retrieval. Kevyn and Nickolov [7] used both the layout and the text features for matching the documents for retrieval.

Jayant et al. [4] propose a method that relies on the patch code words derived from the document images. The code book is learned independently of the class labels of the documents. In the first step, the images are recursively partitioned both in horizontal and vertical direction for modeling spatial relationships. Subsequently, a histogram for each partition is calculated that is used for the classification. Following the same idea of developing the code book, another work presented by Jayant et al. [6] build a codebook of SURF descriptors extracted from training images. Then, histograms of codewords are created similar to [4]. A Random Forest classifier is used for classification. The applicability of the approach is shown in the presence of limited data. Chen et al. [5] propose a method based on low-level image features to classify documents. The approach is limited to structured documents. An important point is that one could obtain the registration of two images by matching the feature points. Joutel et al. [23] proposed a method that used curvelet transformation for indexing and querying the documents at different image scales. Their method is designed particularly for large databases of handwritten manuscripts. Kochi and Saitoh [24] used textual descriptions of document images for information extraction from documents. The method is limited to semi-structured documents and assumes a pre-defined knowledge is available for the document classes. Reddy and Govindaraju [25] used binary images for the classification of the documents. They use pixel information and calculate pixel densities. They used k-means clustering supported by adaptive boosting. The method is evaluated on the benchmark NIST scanned special tax form databases 2 and 6.

The pioneering work that performed document classification using CNNs used a rather shallow network for classification [19]. Nevertheless, the proposed approach outperformed structural similarity based methods and shows the potential of automatic feature learning for document classification using CNNs. The reason may be that deep networks require a lot of data for training and at that time the standard challenging dataset consisted of only 3, 482 images. Afzal et. al. [18] and Harley et. al. [20] provided a breakthrough when they showed
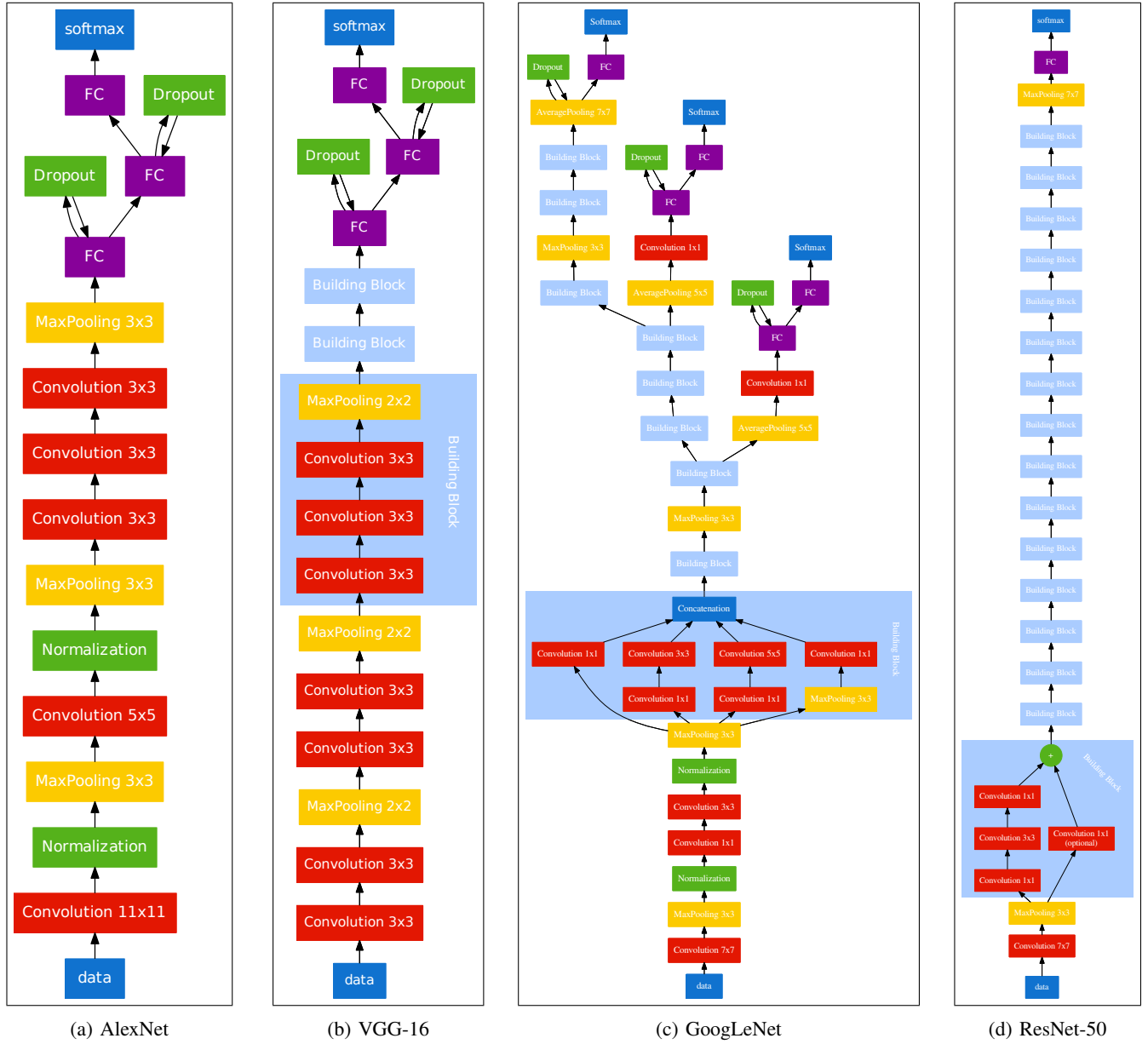
Fig. 3: Deep CNN architectures used in this work

that it is possible to use transfer learning and the features that are learned from general (daily life) images can be used for the classification of document images [18]. They achieved a significant improvement over CNN based methods that were the state-of-the-art at that time. Another notable contribution by Harley et. al. [20] was that they introduced a dataset consisting of 400,000 documents divided into 16 classes. This allowed for the evaluation of deep neural networks using a significant amount of data.

The state-of-the-art in deep CNNs has advanced significantly in recent years and there has been no comprehensive study regarding the impact of deep architectures for document classification. Moreover, there is no study that explores transfer learning from document images and also there is no report of

the impact of the amount of training images. The presented work takes into account these issues and performs a comprehensive set of experiments to fill the gaps that exist. Eventually, this study leads to an approach that can reduce the error by more than half and therefore provides another leap forward in the domain of document image classification.

III. Deep Convolutional Neural Networks

This section briefly presents the deep CNN architectures used in this work. Furthermore, the image preprocessing and training details are described.

A. Network Architectures

The deep CNN architectures used in this paper are well known in the domain of object recognition but are not used

frequently for document image classification. The networks are of very different nature (cf. Fig. 3).

1) AlexNet: AlexNet [26] is the eight-layer CNN that won the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) in 2012 [21] by a large margin. It employs five convolutional layers with optional pooling and local response normalization. These are then followed by three fully-connected layers and a softmax classifier (cf. Fig. 3a).

2) VGG-16: VGG-16, as the name suggests is a 16-layer CNN [27]. Unlike AlexNet, it uses only convolutional filters of size 3×3. Just like AlexNet, it has a straightforward architecture, but with 13 convolutional layers and 3 fully connected layers (cf. Fig. 3b) it is quite a bit deeper and has a repetitive pattern of layers. This architecture has won the localization category of the ILSVRC 2014.

3) GoogLeNet: GoogLeNet, just like VGG-16, won a category of the ILSVRC 2014, namely the classification category [28]. The architecture of this network, however, is a bit more sophisticated (cf. Fig. 3c). Unlike AlexNet and VGG-16, it is not just a stack of Convolution layers and Pooling layers, but rather a stack of building blocks, which themselves consist of Convolution and Pooling layers. It is therefore a Network-in-Network approach [29]. Due to its high depth, the network employs three softmax classifiers during training, to enable efficient backpropagation of the error. At test time, the two auxiliary classifiers are discarded.

4) Resnet-50: ResNets are a family of very deep CNN architectures which make use of residual connections [30] to overcome the challenge of efficient error backpropagation. ResNet-50 is a variant of the network with 50 layers, which, as in GoogLeNet, are grouped in building blocks (cf. Fig. 3d). An even deeper variant with 152 layers won the ILSVRC classification task in 2015. Interestingly, despite its increased depth, the network has fewer parameters to fit than VGG-16.

B. Preprocessing

As the networks used in this paper require images of a fixed size as input, we first downscale all images to the expected input size of the networks. For AlexNet, the images are resized to 227×227 pixels, for the other networks the images are resized to 224×224 pixels. Typically, when training CNNs, the training data is augmented by resizing the images to a larger size, e.g. 256×256 pixels and then cropping random patches of these images in the size of the network input. This approach has shown to be effective for real-world image classification [26]. In real-world images, the objects are typically close to the center of the image and therefore always contained in the random crops. However, the most discriminative parts of document images are not always close to the center of the image but reside in the outer regions, e.g. the head of a letter. Therefore, we do not enlarge our training dataset in this way but train solely with images containing the entire document.

After resizing the images, we compute the mean pixel values of the training images and subtract them from all images to center the training data.

As a last preprocessing step, we convert the grayscale images to RGB images by simply copying the pixel values of the single-channel images to three channels.

C. Training Details

We train all networks using stochastic gradient descent with a momentum of 0.9 and a learning rate that is updated every iteration to

$$lr = initial\_lr * \left(1 - \frac{iter}{max\_iter}\right)^{0.5} \qquad (1)$$

The initial learning rate is set to a value between 0.01 and 0.0001 depending on the network architecture, the training dataset and the weight initialization.

The number of training epochs depends on the task and ranges between 40 and 80 epochs.

IV. EXPERIMENTS AND RESULTS

A. Datasets

To evaluate the performance of the deep neural networks presented in section III, two datasets are used. First, we train a variety of networks on the Ryerson Vision Lab Complex Document Information Processing (RVL-CDIP) dataset [20]. This dataset consists of $400,000$ labeled document images from 16 classes. The dataset is already split into a training dataset which contains $320,000$ images and a validation and a test dataset which each contain $40,000$ images.

Secondly, we use the Tobacco-3482 dataset [6] to evaluate the performance of the deep CNNs and to investigate to which extent transfer learning from the first dataset is applicable. The Tobacco-3482 dataset contains $3,482$ images from ten document classes.

Both datasets are quite similar and there even exists some overlap. Therefore, at the transfer learning experiments, we pretrain the networks not on the full RVL-CDIP dataset, but only on the images that are not contained in the Tobacco-3482 dataset. Thus, the networks are pretrained on only $319,784$ training images.

B. Evaluation

For the RVL-CDIP dataset, we just train the networks and report the top-1 accuracy achieved on the test set. Since the Tobacco-3482 dataset is so small, we use a slightly more sophisticated evaluation technique to get an expected accuracy and to avoid unrepresentative results due to random initialization or a specific dataset split. To come up with a robust estimate of how well the networks perform, we split the dataset such, that 10 to 100 images per class are used for training while the rest are for testing. The training dataset is again split with an $80/20$ ratio, so that 20% of the training data are used for validation. For each split size, we randomly create ten dataset partitions and report the median accuracy achieved by the networks. This is similar to the evaluation scheme that was also used by Kang et al. [19] and Kumar et al. [6] and allows for a fair comparison with their approaches.

TABLE I: Performance of the networks on the Tobacco-3482 dataset with 100 training samples per class and different weight initializations.

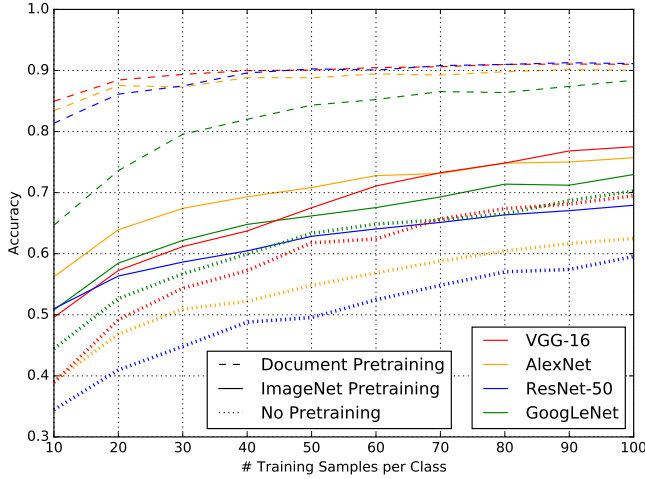|	Document Pretraining	ImageNet Pretraining	No Pretraining
AlexNet	90.04 %	75.73 %	62.49 %
GoogLeNet	88.40 %	72.98 %	70.28 %
VGG-16	91.01 %	77.52 %	69.50 %
Resnet-50	91.13 %	67.93 %	59.55 %

Fig. 4: Mean accuracy achieved on the Tobacco-3482 dataset. The dashed graphs represent the networks that are pretrained on the RVL-CDIP dataset, the solid lines represent the network with ImageNet pretraining and the dotted lines show the network accuracy when trained from scratch.

C. Results on Tobacco-3482

We have trained the four deep CNNs described in Section III on the two datasets with different weight initializations to investigate the benefits of transfer learning. As shown in Table I and Fig. 4 which correspond to the achieved performance on the Tobacco-3482 dataset, transfer learning does improve the classification performance significantly. When the networks are pretrained on a similar dataset, the accuracy achieved on the final dataset is higher than 90 % and already with as little training data as 10 samples per class, we could outperform the current state-of-the-art which achieves only 77.6 % [18].

D. Analysis

We also compare the networks with ImageNet initialization against randomly initialized networks and find that even though the images are substantially different (cf. Fig. 1), it helps to use pretrained models. Fortunately, there are models which are pretrained on ImageNet available online for many architectures, including the four networks used in this work. So, in case there is no large document dataset available for pretraining, one can and should always resort to using an ImageNet pretrained model for finetuning.

TABLE II: Performance of the networks on the RVL-CDIP dataset with different weight initializations.

	ImageNet Pretraining	No Pretraining
AlexNet	88.60 %	88.19 %
GoogLeNet	89.02 %	88.60 %
VGG-16	90.97 %	89.41 %
Resnet-50	90.40 %	89.24 %

Depending on the amount of available training data, AlexNet and VGG-16 are the best choices when finetuning the networks from models that were pretrained on ImageNet (cf. Fig. 4). When pretrained on the RVL-CDIP dataset, GoogLeNet is significantly worse than the other networks, especially for a small amount of training data.

E. Results on RVL-CDIP

On the large-scale RVL-CDIP dataset, all networks achieve very good results (cf. Table II) with the VGG-16 performing best at an accuracy of 90.97 %. The current state-of-the-art on this dataset only achieves an accuracy of 89.8 %, thus we could decrease the relative error by more than 11 % by simply using a different network architecture. Note, that even though the training dataset is quite large, all of the networks still benefit from Imagenet pretraining.

On average, VGG-16 performs very well on all experiments performed in this work. As can be seen in Fig. 5 which shows the confusion matrix of a trained VGG-16 network, even the classes that were pointed out to be hard by Afzal et al. [18], get significant performance boosts.

V. CONCLUSION AND FUTURE WORK

The outcome of the study brings insights both for the deep neural network architectures and the amount of required training data. The proposed approach of training on document images and then finetuning for a document based dataset improved the error by 60 %. We show that the random initialization performs worst and initialization based on document images performs best. Furthermore, on the large-scale RVL-CDIP dataset, VGG-16 outperforms the other networks. Finally, a relative error reduction of 11.5 % compared to the state-of-the-art is achieved. Future work may evaluate recurrent neural networks or a combination of convolutional and recurrent neural networks to improve the performance further.
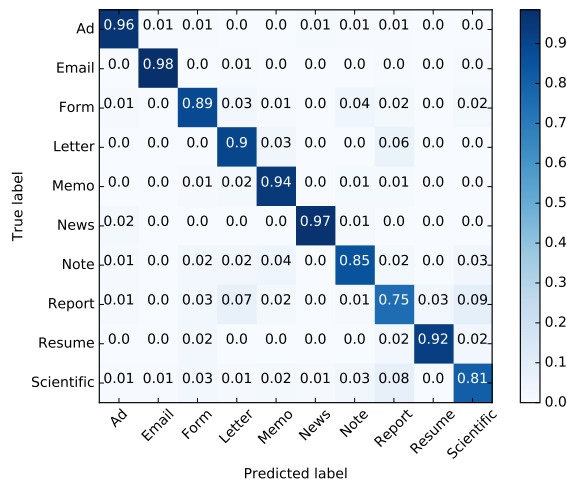
Fig. 5: Confusion Matrix by a VGG-16 network trained on the Tobacco-3482 dataset.

REFERENCES

[1] A. Dengel and F. Dubiel, "Clustering and classification of document structure-a machine learning approach," in *ICDAR*, 1995, pp. 587–591.

[2] Y. Byun and Y. Lee, "Form classification using dp matching," in *ACM Symposium on Applied Computing*, New York, USA, 2000, pp. 1–4.

[3] C. Shin and D. S. Doermann, "Document image retrieval based on layout structural similarity." in *IPCV*, 2006, pp. 606–612.

[4] J. Kumar, P. Ye, and D. S. Doermann, "Learning document structure for retrieval and classification." in *ICPR*. IEEE, 2012, pp. 1558–1561.

[5] S. Chen, Y. He, J. Sun, and S. Naoi, "Structured document classification by matching local salient features," in *ICPR*, Nov 2012, pp. 653–656.

[6] J. Kumar, P. Ye, and D. Doermann, "Structural similarity for document image classification and retrieval," *Pattern Recognition Letters*, vol. 43, pp. 119 – 126, 2014.

[7] K. Collins-thompson and R. Nickolov, "A clustering-based algorithm for automatic document separation," 2002.

[8] M. Z. Afzal, J. Pastor-Pellicer, F. Shafait, T. M. Breuel, A. Dengel, and M. Liwicki, "Document image binarization using lstm: A sequence learning approach," in *Proceedings of the 3rd International Workshop on Historical Document Imaging and Processing*. ACM, 2015, pp. 79–84.

[9] J. Pastor-Pellicer, S. España-Boquera, F. Zamora-Martínez, M. Z. Afzal, and M. J. Castro-Bleda, "Insights on the use of convolutional neural networks for document image binarization," in *International Work-Conference on Artificial Neural Networks*. Springer International Publishing, 2015, pp. 115–126.

[10] J. Pastor-Pellicer, M. Z. Afzal, M. Liwicki, and M. J. Castro-Bleda, "Complete system for text line extraction using convolutional neural networks and watershed transform," in *Document Analysis Systems (DAS), 2016 12th IAPR Workshop on*. IEEE, 2016, pp. 30–35.

[11] M. Seuret, M. Alberti, R. Ingold, and M. Liwicki, "Pca-initialized deep neural networks applied to document image analysis," *arXiv preprint arXiv:1702.00177*, 2017.

[12] M. Liwicki, A. Graves, H. Bunke, and J. Schmidhuber, "A novel approach to on-line handwriting recognition based on bidirectional long short-term memory networks," in *9th International Conference on Document Analysis and Recognition (ICDAR)*, 2007, pp. 367–371.

[13] T. M. Breuel, A. Ul-Hasan, M. A. Al-Azawi, and F. Shafait, "High-performance OCR for printed English and Fraktur using LSTM networks," in *Document Analysis and Recognition (ICDAR), 2013 12th International Conference on*. IEEE, 2013, pp. 683–687.

[14] R. Ahmad, M. Z. Afzal, S. F. Rashid, M. Liwicki, and T. Breuel, "Scale and rotation invariant ocr for pashto cursive script using mdlstm network," in *Document Analysis and Recognition (ICDAR), 2015 13th International Conference on*. IEEE, 2015, pp. 1101–1105.

[15] S. B. Ahmed, S. Naz, M. I. Razzak, S. F. Rashid, M. Z. Afzal, and T. M. Breuel, "Evaluation of cursive and non-cursive scripts using recurrent neural networks," *Neural Computing and Applications*, vol. 27, no. 3, pp. 603–613, 2016.

[16] S. Ahmed, M. I. Malik, M. Z. Afzal, K. Kise, M. Iwamura, A. Dengel, and M. Liwicki, "A generic method for automatic ground truth generation of camera-captured documents," *arXiv preprint arXiv:1605.01189*, 2016.

[17] A. Ul-Hasan, M. Z. Afzal, F. Shafait, M. Liwicki, and T. M. Breuel, "A sequence learning approach for multiple script identification," in *Document Analysis and Recognition (ICDAR), 2015 13th International Conference on*. IEEE, 2015, pp. 1046–1050.

[18] M. Z. Afzal, S. Capobianco, M. I. Malik, S. Marinai, T. M. Breuel, A. Dengel, and M. Liwicki, "Deepdocclassifier: Document classification with deep convolutional neural network," in *Document Analysis and Recognition (ICDAR), 2015 13th International Conference on*. IEEE, 2015, pp. 1111–1115.

[19] Le Kang, Jayant Kumar, Peng Ye, Yi Li, and David Doermann, "Convolutional Neural Networks for Document Image Classification," in *ICPR*, 2014.

[20] A. W. Harley, A. Ufkes, and K. G. Derpanis, "Evaluation of deep convolutional nets for document image classification and retrieval," in *13th International Conference on Document Analysis and Recognition (ICDAR), 2015*. IEEE, 2015, pp. 991–995.

[21] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[22] A. Bagdanov and M. Worring, "Fine-grained document genre classification using first order random graphs," in *ICDAR*, 2001, pp. 79–83.

[23] G. Joutel, V. Eglin, S. Bres, and H. Emptoz, "Curvelets based queries for cbir application in handwriting collections," in *ICDAR*, 2007, pp. 649–653.

[24] T. Kochi and T. Saitoh, "User-defined template for identifying document type and extracting information from documents," in *ICDAR*, Sep 1999, pp. 127–130.

[25] K. V. U. Reddy and V. Govindaraju, "Form classification," in *DRR'08*, 2008, pp. −1–1.

[26] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105.

[27] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[28] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1–9.

[29] M. Lin, Q. Chen, and S. Yan, "Network in network," *arXiv preprint arXiv:1312.4400*, 2013.

[30] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.