Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

DocXClassifier: Towards a Robust and Interpretable Deep Neural Network for Document Image Classification

SAIFULLAH SAIFULLAH[12], STEFAN AGNE[13], ANDREAS DENGEL[12], AND SHERAZ AHMED[13]

[1]German Research Center for Artificial Intelligence (DFKI) 67663 Kaiserslautern, Germany
e-mail: {saifullah.saifullah,stefan.agne,andreas.dengel,sheraz.ahmed}@dfki.de
[2]Department of Computer Science, University of Kaiserslautern-Landau, Erwin-Schrödinger-Straße 52, 67663 Kaiserslautern, Germany
[3]DeepReader GmbH, 67663 Kaiserlautern, Germany

Corresponding author: Saifullah (e-mail: saifullah.saifullah@dfki.de).

This work was supported by the BMBF projects SensAI (BMBF Grant 01IW20007)

ABSTRACT Model interpretability and robustness are becoming increasingly critical today for the safe and practical deployment of deep learning (DL) models in industrial settings. As DL-backed automated document processing systems become increasingly common in business workflows, there is a pressing need today to enhance interpretability and robustness for the task of document image classification, an integral component of such systems. Surprisingly, while much research has been devoted to improving the performance of deep models for this task, little attention has been given to their interpretability and robustness. In this paper, we aim to improve upon both aspects and introduce DocXClassifier, an inherently interpretable deep document classifier that not only achieves significant performance improvements over existing approaches in image-based document classification, but also holds the capability to simultaneously generate feature importance maps while making its predictions. Our approach attains state-of-the-art performance in image-based classification on two popular document datasets, RVL-CDIP and Tobacco3482, with top-1 classification accuracies of 94.17% and 95.57%, respectively. Additionally, it sets a new record for the highest image-based classification accuracy on Tobacco3482 without transfer learning from RVL-CDIP, at 90.14%. In addition, our proposed training strategy demonstrates superior robustness compared to existing approaches, significantly outperforming them on 19 out of 21 different types of novel data distortions, while achieving comparable results on the remaining two. By combining robustness with interpretability, DocXClassifier presents a promising step towards the practical deployment of DL models for document classification tasks.

INDEX TERMS Document Image Classification, Explainable Document Classification, Model Interpretability, Inherent Interpretability, Model Robustness, Corruption Robustness

I. INTRODUCTION

In recent years, deep learning (DL) has made significant breakthroughs in the field of document analysis demonstrating exceptional performance on a range of tasks such as document classification [1]–[3], key information extraction (KIE) [3], [4], and layout analysis [5]. Despite these performance gains, however, there remain two major challenges that continue to hinder the safe and secure deployment of such DL-based systems in real-world scenarios: their inherent black-box nature [6], [7] and their poor robustness to out-of-distribution (OOD) data [8]–[11].

The lack of transparency in DL-based automated decision-making is particularly concerning. A number of recent studies have demonstrated that DL models are prone to learning biases from the data [12]–[14], resulting in unfair decisions or discrimination against individuals from certain racial or gender identities [6], [7]. Such biases can have catastrophic effects in the context of document classification. For instance, a DL-based document classifier that categorizes applicant resumes as acceptable or otherwise could potentially learn to discriminate against women or minority groups. For these reasons, model interpretability is crucial, as it can help

identify biases in the data and provide insights into the model's decision-making process, ultimately enabling their safe deployment [13], [14].

Besides fairness and ethical considerations, model robustness is another important factor for the scalable and efficient deployment of DL-based systems in practical settings [10]. Recent studies have shown that DL-based systems perform poorly when faced with minor distribution shifts in the data [9], [15], [16], even when trained with a number of data augmentations [10]. Such distribution shifts are commonly occurring in real-world scenarios [10], especially in the document domain [11], [17], where documents are often corrupted with novel distortions at test time, such as the addition of noise, blur, ink-bleed, or stain marks [11], [18]. One straightforward example is mobile-captured documents, which are commonly used by the end users but may end up with transformations or noise due to varying lighting conditions [19]. Such OOD data if encountered at test time may result in model failure modes [11]. Note that transparency of decision-making is also crucial in this context to identify the potential reasons behind these failures.

In recent years, a wide variety of approaches have been proposed that attempt to explain the predictions of black-box DL models [6], [20]–[24]. In the image domain, post-hoc attribution-based approaches [20]–[22] are the most popular, which generate feature importance maps to identify the areas in the image that were most important to the model's prediction. Despite their widespread use, however, they are not without limitations, such as their costly processing times [25], [26] and potentially unfaithful explanations [27], [28]. On the other hand, numerous studies have also explored different strategies to enhance model robustness to OOD data [10], [29], [30], with data-augmentation strategies [31]–[34] being particularly popular in this regard.

In this paper, we tackle the challenges of both model interpretability and robustness in the context of document image classification, which is a core component of modern document processing pipelines [17], [35]–[37]. Despite a significant amount of research dedicated to improving the performance of DL models for this task [1], [3], [17], [36], [38], we found that research paying attention to their robustness and interpretability is relatively scarce [39]. In this work, therefore, we focus on all three aspects: performance, robustness and interpretability, and propose DocXClassifier, a high-performing and inherently-interpretable deep convolutional neural network (ConvNet) for document image classification. Our approach involves modifying the architecture of the recently introduced ConvNeXt model [40] with an additional attention-pooling mechanism that allows it to attribute importance to image features in a single forward pass, removing the need to use costly post-hoc attribution methods [25], [26] for generating explanations. In order to achieve improved performance and robustness, we devise a two-stage training strategy and apply several data augmentation strategies, including RandomAugment [31], CutMix [41], Mixup [32], and RandomErasing [33], in combination with modern training approaches, such as Label Smoothing [42], Exponential Moving Average (EMA), and LayerDecay [40]. To the best of authors' knowledge, this is the first work that explores the combination of all these approaches in this context. The overall contributions of this paper are two-fold:

- We propose an inherently-interpretable deep ConvNet for document image classification, which has the capability to generate feature-importance maps for input images in a single forward pass. To the best of authors' knowledge, this is the first work in this direction.
- Our proposed approach not only achieves a state-of-the-art performance in image-based document classification, but also outperforms some existing multimodal approaches. We demonstrate the effectiveness of our method on two well-known document classification benchmark datasets, RVL-CDIP [17] and Tobacco3482. On RVL-CDIP [17], our approach achieves an accuracy of 94.17%, significantly surpassing the previous state-of-the-art, which had an accuracy of 92.31%. On the Tobacco3482 dataset, we trained our models with and without RVL-CDIP [17] pre-training and obtained accuracies of 95.57% and 90.14%, significantly outperforming the previous state-of-the-art methods that yielded accuracies of 94.04% and 85.9% respectively.
- Our proposed training strategy demonstrates superior robustness to existing state-of-the-art document image classification approaches, outperforming them on 18 out of 21 different novel distortion types, while achieving comparable results on the remaining ones.

II. RELATED WORK

A. DOCUMENT IMAGE CLASSIFICATION

The subject of document image classification has been extensively explored in the past few decades. Earlier attempts to classify document images were either based on traditional computer vision techniques, such as feature matching [43], [44] or classical machine learning approaches, such as K-Nearest Neighbors [45], or Random Forest Classifiers [44]. For a detailed overview of these approaches, we refer the reader to a related survey [35].

With the advent of deep learning, the field of document image classification has experienced a major transformation. Kang *et al.* (2014) [46] were the first to investigate Convolutional Neural Networks (ConvNets) in the context of document image classification and demonstrated significant performance improvements over classical feature engineering approaches with just a shallow network. Afzal *et al.* (2015) [47] and Harley *et al.* (2015) [17] in parallel explored the potential of transfer learning in combination with deep networks in their work, showing that fine-tuning models already pre-trained on the large-scale ImageNet [48] dataset can lead to significantly better feature representations and consequently better performance. Afzal *et al.* (2017) [36] later extended these works to much deeper ConvNets achieving breakthrough performance improvements in document image classification. In a more recent approach, Ferrando

et al. (2020) [1] investigated parallel training techniques on EfficientNet [49] models and achieved a new peak performance for image-based document classification. Vision Transformers (ViTs) [11], [50], [51] have also gained some attention in document image classification [52], however, more work is needed before they can match the performance of the latest ConvNets in this domain.

Recently, there has been an increased emphasis on multimodal classification techniques [2], [53], [54], in which document images are preprocessed to extract the textual content using stand-alone optical-character-recognition (OCR) software, and then visual, textual, and other layout features are used together for classification. Initial work in this area focused on using two separate deep network streams [2], [37] for multimodal classification. Recently, however, large-scale transformer-based document foundation models such as LayoutLM [3], [55], TILT [4], and UDOP [56] have become more popular that simultaneously use visual, textual, and layout features as input and produce an integrated multimodal document representation for resolving various document understanding tasks. These approaches, however, require extensive pre-training with large amounts of document data.

It is worth mentioning that since multimodal approaches require a pre-processing step that uses a standalone OCR software to extract the textual information from the documents, their performance is heavily dependent on the performance of the OCR system which not only adds additional computational overhead but also additional complexity in the system. Moreover, the robustness aspects of such models are rarely discussed in the literature even though the OCR systems could be severely affected by novel distortions in the document data which could in turn degrade the performance of such systems.

B. EXPLAINABLE AI (XAI)

The field of eXplainable AI (XAI) has attracted considerable attention in recent years, with numerous approaches developed to explain the predictions of black-box artificial intelligence (AI) models [6], [20]–[22]. These approaches range from model-agnostic gradient-based methods such as GradCAM [21], IntegratedGradients [57], or DeepLIFT [58], to perturbation-based approaches such as LIME [20], or SHAP [22]. Gradient-based approaches utilize model gradients to determine the importance of each input feature. Perturbation-based approaches, on the other hand, perturb the input and measure the impact of perturbation on the model's prediction in order to determine feature importance. While such attribution-based methods have been widely adopted to generate explanations for DL systems on various tasks [6], [25], they also have some potential limitations. These include the requirement of significant domain knowledge for interpreting explanations [27], [28], costly computation [25], [26], potentially unfaithful explanations [59], and the challenges of selecting the appropriate perturbation model when applying perturbation-based approaches [26].

In light of these limitations, many researchers have recently argued against the use of model-agnostic approaches [27], [28], and have brought forward the idea that deep learning models should be made interpretable by design [27]. This has led to the development of prototype-based approaches [60], and explainable deep neural networks [61], [62]. In this work, we adhere to the same spirit and focus on interpretability by design instead of relying on model-agnostic approaches to generate explanations. Besides those discussed, there are several other interpretability approaches that have been proposed in the literature in recent years, such as concept-based explanations [63], and generative models-based explanations [23], [24].

C. MODEL ROBUSTNESS

Model robustness has also been the subject of extensive research in the past few years, which has been mainly divided into two broad categories: (1) adversarial robustness [64], which deals with robustness against adversarial attacks, and (2) corruption robustness [10], [30], which deals with robustness against novel corruptions in data, such as blur, noise, occlusions, and pose variations. In this work, we are primarily concerned with improving corruption robustness since it is more relevant to real-world deployment scenarios. Therefore, further related work pertaining to this aspect is subsequently discussed.

In recent years, many advanced data augmentation strategies [33], [34], [65] have been proposed that allow DL models to learn better features, reduce model overfitting, and improve robustness against out-of-distribution (OOD) data. Techniques such as AutoAugment [65] and RandAugment [31] search for the optimal data augmentation policy to augment input images, which are then used to train the model for enhanced robustness. Random Erasing [33] is a data augmentation techniques that randomly erases parts of the image and has been demonstrated to improve robustness against partially occluded images. More recent strategies like CutMix [41], Mixup [32], and AugMix [34], on the other hand, augment input images by mixing samples from multiple classes and have been demonstrated to be very effective for enhancing overall robustness. These techniques are also often accompanied by Label Smoothing [42], which regularizes the models by preventing them from predicting the output labels too confidently.

Hendrycks *et al.* (2019) [10] recently proposed robustness benchmark datasets for standard ImageNet [48] datasets in order to evaluate the robustness characteristics of deep neural networks. Subsequent research has extensively used these datasets to evaluate and improve the robustness of deep learning models using various strategies [30], [34]. Taking inspiration from this, Saifullah *et al.* (2022) [11] recently introduced two robustness benchmark datasets, namely RVL-CDIP-D and Tobacco3482-D, designed for the document domain. These datasets were generated by applying 21 different types of novel distortions to generate out-of-distribution counterparts for the RVL-CDIP [17] and Tobacco3482 doc-

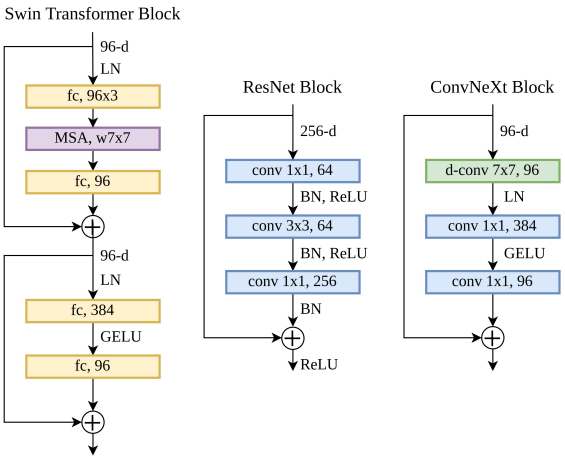


FIGURE 1. Block configurations of ConvNext, ResNet and Swin Transformer are shown for comparison.

ument datasets, respectively. Furthermore, the datasets were used to evaluate the robustness characteristics of several existing state-of-the-art document image classification models. In this work, we use both of these datasets for evaluating the robustness of our proposed models.

III. METHODOLOGY

In this section, we present the details of our proposed architecture, the data augmentation techniques investigated for improving model robustness, and the training strategies that were utilized in our study.

A. DOCXCLASSIFIER: MODEL ARCHITECTURE

Our work is an extension over the recently proposed ConvNeXt [40] model that is not only heavily inspired by the state-of-the-art ViTs, but can also outperform them in the domain of natural image classification. In particular, ConvNeXt [40] was developed by making various design modifications to the standard ResNet model [66]—modifications inspired by both modern convolutional neural networks (ConvNets) and the recently introduced Swin Transformers [67], a variant of ViTs. Since ResNet-50 has been previously investigated for document classification by Afzal et al. (2017) [36], we will briefly explain the modifications in the following sections to emphasize the distinctions between our work and theirs. The design changes between ConvNeXt and ResNet model fall into two main categories: Macro Design and Micro Design.

Macro Design. ConvNeXt uses a stage compute ratio of 1:1:3:1 as compared to $1{:}1\frac{1}{3}{:}2{:}1$ in ResNet-50, which is directly inspired by the Swin Transformers [67]. Another important difference is that ConvNeXt uses a Patchify layer [40], as is common in ViTs [50] for initial downsampling of images. The standard ResNet model uses a 7x7 convolutional layer followed by a max-pooling layer for this purpose, whereas the Patchify layer in ConvNeXt is implemented with a non-overlapping convolutional layer of kernel size 4x4 and a stride 4. Compared to ResNet-50, ConvNeXt uses depth-wise convolutions [68] instead of using standard convolution operations as in ResNet-50. In addition, the inverted bottleneck was introduced in each block, but with the convolutional layers shifted up in order, a design decision again inspired by Transformers, where the multi-self-attention blocks are generally placed before the fully connected layers.

Micro Design. Some minor architectural changes were also made for improving performance. For example, the ReLU activations were replaced with GELU activations, which are commonly used in latest Transformer models. The total number of activations were reduced so that there was only a single activation function at the end of each block. The total number of normalization layers were also reduced and batch normalization was removed in favor of layer normalization. Finally, the initial residual block in ResNet was removed and instead a separate downsampling layer, followed by layer normalization, was added between each stage to mimic the Swin Transformers [67].

Attention-Based Pooling. Although a considerable number of design changes were made, the resulting ConvNeXt model is just another ConvNet without any sophisticated components, as can be observed in Fig. 1. In this work, we modify the existing ConvNeXt model with an attention-based mechanism to force its predictions to be based on different regions of the image, ultimately making it inherently interpretable. Since the original ConvNeXt models are simply ConvNets, they are not capable of generating feature importance maps out-of-the-box and are generally augmented with a linear layer with global average pooling to perform the classification task. We modify the ConvNeXt architecture by substituting the global average pooling of ConvNeXt with an attention-based pooling mechanism [62] as shown in Fig. 2. In particular, the attention-based pooling mechanism employs a query class token to aggregate the output feature map vectors of the model as a weighted sum based on their similarity to a trainable class (CLS) vector of dimension d, which is similar to a class token in transformers [69]. Whereas, the similarity is computed using the standard scaled dot-product attention operation [69]:

$$\text{Attn}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V \tag{1}$$

where K, Q, and V represent the query, key, and value matrices of the attention layer, respectively, and d_k represents the feature dimension of the kth attention head. With

TABLE 1. Number of channels and blocks per stage for different variants of the DocXClassifier model.

Model	Channels	Blocks
DocXClassifier-B	(128, 256, 512, 1024)	(3, 3, 27, 3)
DocXClassifier-L	(192, 384, 768, 1536)	(3, 3, 27, 3)
DocXClassifier-XL	(256, 512, 1024, 2048)	(3, 3, 27, 3)

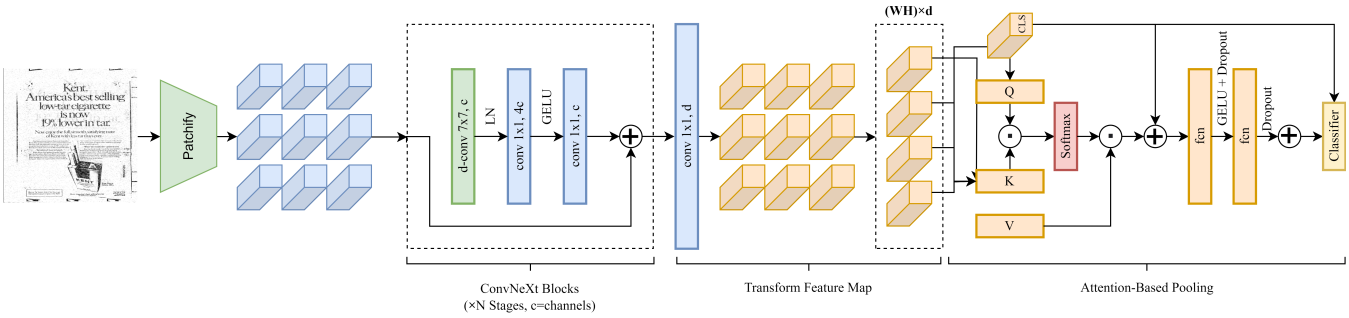


FIGURE 2. Complete configuration of the proposed DocXClassifier model. The base ConvNeXt model is used as the backbone for generating a feature map for the input image. The feature map is then transformed and fed into a attention-based pooling mechanism to force the predictions of the models to be based on different regions of the image. Finally, a linear classification head is used to generate the class prediction scores.

the attention mechanism applied only once, using a single softmax operation, the model is essentially forced to assign importance to certain feature vectors for each specific class. The resulting aggregated vector is then added to the CLS vector and processed by a feed-forward network. Finally, a linear classification head is used to perform the classification.

Note that before applying the attention-mechanism, we also first apply point-wise convolution to the output feature map of the model to transform its dimensions from $C \times H \times W$ to $d \times H \times W$ so that it matches the dimensions of the CLS token, and then reshape the output feature map to a $(HW) \times d$ dimension. These HW feature vectors which are then fed to the attention-based pooling mechanism as shown in Fig. 2. This feature transformation step was necessary as the output dimensions of the feature map for different variants of the model can result in different channel dimensions, for instance, a dimension of size 1024 for the DocXClassifier-B variant as shown in Table. 1. If not down-scaled to a fixed dimension, the attention pooling mechanism can result in considerably high number of parameters, especially in case of larger variants. Therefore, to keep the number of parameters low, we kept a fixed embedding dimension of $d = 768$ for both the CLS vector and the final feature maps on which the attention pooling is applied.

The complete model configuration with these modifications is shown in Fig. 2, which we refer to as DocXClassifier. The implementation details of the model can be found at https://github.com/saifullah3396/docxclassifier.git. We define three different variants of the model namely, DocXClassifier-B, DocXClassifier-L, and DocXClassifier-XL, similar to the original ConvNeXt [40] model with different configurations of number of blocks and stages as shown in Table 1.

B. DATA PRE-PROCESSING

Basic pre-processing steps in our approach include converting grayscale images to RGB color space, downscaling the images to a fixed input resolutions of size 224×224 or 384×384 as required, and normalizing the images by subtracting the ImageNet mean $(0.485, 0.456, 0.406)$ and dividing by the ImageNet standard deviation $(0.229, 0.224, 0.225)$, as done in previous works [1], [47].

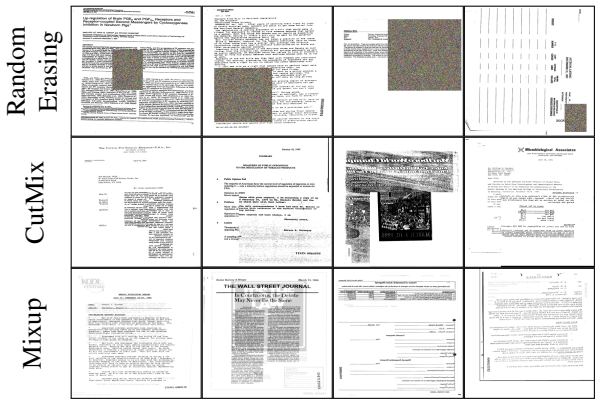


FIGURE 3. Few sample document images generated by applying the data augmentation techniques: RandomErasing [33], CutMix [41], and Mixup [32], respectively. As shown, RandomErasing [33] removes patches from images by replacing them with per-pixel random normal. CutMix [41] spatially merges samples of different classes. Whereas, Mixup [32] merges samples from different classes by overlaying them over each other.

C. DATA AUGMENTATIONS AND GENERALIZATION STRATEGIES

Besides basic pre-processing steps, we explored more advanced data augmentation strategies in this work to improve model generalization and robustness. In many previous works on document image classification [1], [36], [39], we have encountered the common belief that data augmentation techniques developed for natural images cannot be directly applied to document images due to the fundamental differences between these two image types. As a result, these works have typically applied only minor augmentations to document images, such as simple shear transformations [1], [39]. In this work, we demonstrate that employing more aggressive data augmentation techniques can significantly enhance both the performance and robustness of document classification models. Multiple data augmentation techniques were used in combination for this purpose, as outlined below:

- **RandAugment [31]:** We apply the RandAugment [31] [1] approach for document images, which randomly applies various data augmentations such as changing image color, brightness, contrast, sharpness and applying transformations such as translation, rotation, shear.
- **RandomErasing [33]**: We apply RandomErasing [33] which is used to substitute random patches of images with per-pixel random normal distribution with a probability of 0.25.
- **CutMix [41] and Mixup [32]:** We apply CutMix [41] and Mixup [32] techniques over image batches each with a 50% probability. Cutmix [41] generates a new document sample by spatially merging two samples of different classes whereas Mixup [32] merges them by overlaying the two images on top of each other. Both of the techniques also generate soft labels for target classes based on their visibility in the augmented sample.

A few sample outputs of different augmentation strategies applied on document images are shown in Fig. 3. In addition to data augmentation strategies, we also employ multiple model regularization techniques that have not been previously explored in this context. In this work, we apply Stochastic Depth dropout [70] to randomly drop layers of the network for improved generalization, use Label Smoothing [42] to prevent overfitting on the target labels, and utilize Exponential Moving Average (EMA) [40] during model training, all of which result in significant performance improvements in our experiments.

IV. EXPERIMENTS AND RESULTS

A. DATASETS

To evaluate the performance of our proposed approach on document image classification task, we selected two popular document datasets: RVL-CDIP [17] and Tobacco3482. RVL-CDIP [17] is a large-scale document dataset that has been widely used as a benchmark for document image classification in a number of previous works [1], [17], [36], [55]. The dataset consists of 400K labeled document images with 16 class labels and has training, testing, and validation splits of 320K, 40K, and 40K in size, respectively. Tobacco3482, on the other hand, is a smaller dataset with only 3482 labeled document images and 10 document categories, but is still widely popular for the task of document image classification. There is no predefined partitioning for this dataset. Therefore, we prepared the training set by randomly selecting 80% of the samples per class label, resulting in a training and test set of size 2782 and 700, respectively. Since both datasets are subsets of a much larger dataset, there is some overlap between them. Therefore, for all our experiments, we removed the overlapping images from the training set of RVL-CDIP [17], reducing the size of the training set to 319,756.

For robustness evaluation, we use the two document robustness benchmark datasets, RVL-CDIP-D and Tobacco3482-D [11], as discussed in Sec. II. The datasets were generated by applying 21 different types of data distortions to the RVL-CDIP and Tobacco3482 test sets, respectively. The distortions are broadly categorized into 5 different classes: Noise, Digital Corruptions, Blur, Geometric Distortions, and Document-Specific Distortions, each applied to the test set with 5 different severity levels. Overall, the two datasets, RVL-CDIP-D and Tobacco3482-D, contain approximately 4.2M and 73K evaluation samples, respectively.

B. TRAINING DETAILS

In this section, we provide the details about the training strategies used in each of our experiments.

Training on RVL-CDIP. Since transfer learning has already proven to be successful in the field of document image classification [36], instead of training the models from scratch, we initialized them with the ImageNet-22k [48] pre-trained weights and then fine-tuned them on the RVL-CDIP [17] dataset. All models were trained on 4-8 A100 GPUs with DistributedDataParallel (DDP) using the AdamW optimizer and a cosine decay learning rate strategy with no warm-up period. We chose a base learning rate of 8e-4, corresponding to a batch size of 64, and scaled it linearly with different configurations of batch size, varying between 64, 128, and 256. Since the weights of the attention-based pooling stage were initialized from scratch, we found it difficult to train the models DocXClassifier models end-to-end, and therefore we trained them in two steps. First, we fine-tuned the base ConvNeXt models for 30 epochs to achieve the desired classification performance. Then, we froze the weights of the base model, used them to initialize our DocXClassifier variants, and trained only the attention-based pooling stage along with the classifier.

Training on Tobacco3482. On the Tobacco3482 dataset, we trained the models with two different configurations: with RVL-CDIP [17] pre-training and with ImageNet pre-training. In the first configuration, we simply selected the DocXClassifier models that performed best on the RVL-CDIP [17] dataset and further fine-tuned them on the Tobacco3482 dataset. In this case, we used the same training hyperparameters as above, except that we did not apply EMA in this case as it did not seem to yield any improvements. In the second configuration, we followed the same approach as RVL-CDIP [17], initializing the models with the pre-trained weights from ImageNet-22k [48] and then fine-tuning them directly on the Tobacco3482 dataset in a two-step process. The hyper-parameters used in this configuration were the same as those used in RVL-CDIP [17] training, except for the learning rate and the number of epochs which were set to 5e-5 and 90, respectively.

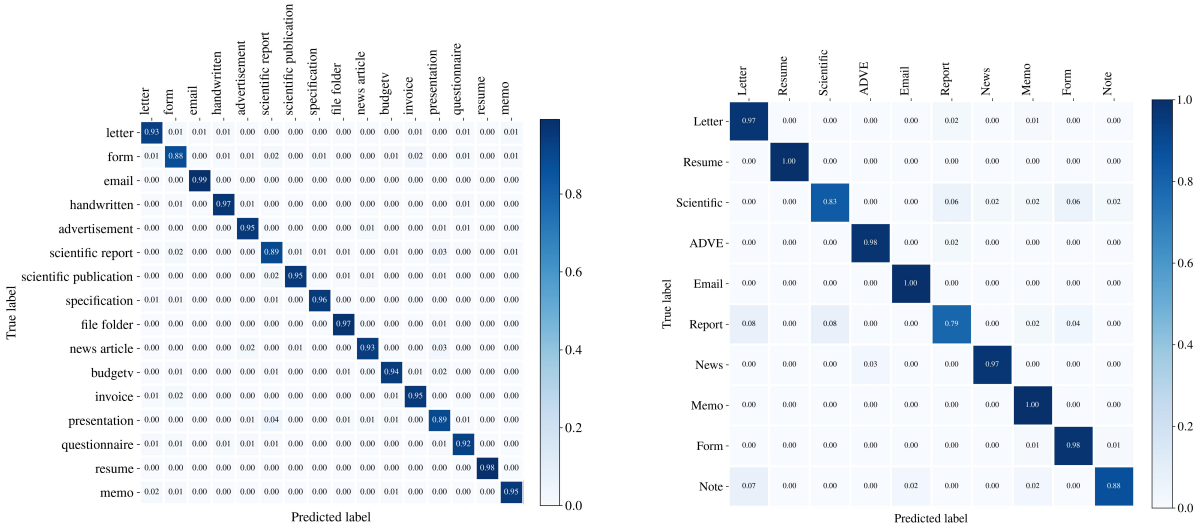
C. PERFORMANCE EVALUATION

Results on RVL-CDIP. Table 2 shows a comparison of the top-1 classification accuracy achieved on the RVL-CDIP [17] and Tobacco-3482 datasets by our approach, previous image-based baseline solutions, and several multimodal approaches that use either text, layout, or both in addition to image

[1] https://github.com/rwightman/pytorch-image-models

TABLE 2. A comparison of the top-1 classification accuracy of different approaches on the RVL-CDIP [17] and Tobacco3482 datasets.

Modality	Model	Inference Time [ms]	# of Parameters	Domain-specific pre-training	RVL-CDIP	Tobacco3482 (RVL-CDIP pre-training)	Tobacco3482 (ImageNet pre-training)
Image	Holistic CNN (Harley *et al.*, 2015 [17])	-	-		89.80%	-	-
	AlexNet (Afzal *et al.*, 2017 [36])	1.1	57M		88.60%	90.04%	75.73%
	GoogleNet (Afzal *et al.*, 2017 [36])	1.2	5.6M		89.02%	88.40%	72.98%
	ResNet-50 (Afzal *et al.*, 2017 [36])	1.1	23.5M		90.40%	91.13%	67.93%
	VGG-16 (Afzal *et al.*, 2017 [36])	1.3	134M		90.97%	91.01%	77.52%
	Stacked CNN Single (Das *et al.*, 2018 [71])	-	-		91.11%	-	-
	Stacked CNN Ensemble (Das *et al.*, 2018 [71])	-	-		92.21%	-	-
	EfficientNet (Ferrando *et al.*, 2020 [1])	2.3	17.6M		92.31%	94.04%	85.99%
	DocXClassifier-B/384 (Ours)	6.53	95.4M		**94.00%**	**95.29%**	**87.43%**
	DocXClassifier-L/384 (Ours)	10.0	204M		**94.15%**	**95.57%**	**88.43%**
	DocXClassifier-XL/384 (Ours)	16.1	356M		**94.17%**	**95.43%**	**90.14%**
Multimodal	MobileNetV2+Text (Audebert *et al.*, 2019 [2])	-	-		90.60%	-	87.80%
	EfficientNet + BERT (Ferrando *et al.*, 2020 [1])	-	127.6M		-	94.90%	89.47%
	LadderNet (Sarkhel *et al.*, 2019 [72])	-	-		92.77%	82.78%	-
	Multimodal Ensemble (Dauphinee *et al.*, 2019 [73])	-	-		93.07%	-	-
	Multimodal GCN (Xiong *et al.*, 2021 [54])	-	49M		93.45%	-	-
	$\text{LayoutLM}_{\text{BASE}}$ (Xu *et al.*, 2020 [55])	-	160M	✓	94.42%	-	-
	$\text{TILT}_{\text{LARGE}}$ (Powalski *et al.*, 2021 [4])	-	780M	✓	95.52%	-	-
	EfficientNet+BERT (Kanchi *et al.*, 2022 [38])	-	197M		95.48%	**95.7%**	**90.3%**
	$\text{LayoutLMv2}_{\text{LARGE}}$ (Xu *et al.*, 2021 [3])	-	426M	✓	95.64%	-	-
	$\text{NasNet}_{\text{Large}}$+$\text{BERT}_{\text{BASE}}$ (Bakkali *et al.*, 2020 [74])	-	197M		**97.05%**	-	-



(a) RVL-CDIP

(b) Tobacco3482

FIGURE 4. The confusion matrices for the DocXClassifier-XL model (with RVL-CDIP [17] pre-training in the case of Tobacco3482) are shown for the two datasets RVL-CDIP [17] and Tobacco3482.

data for classification. As can be seen from the table, our best performing model DocXClassifier-XL achieved 94.17% accuracy on the RVL-CDIP [17] dataset, outperforming all previous image-based methods by a significant margin of +1.86%. It is interesting to note that even our lightest variant DocXClassifier-B achieved a comparable accuracy of 94.00%, and performed significantly better than all existing image-based models as well as some of the more sophisticated multimodal approaches [54], [72], [73], thus representing a good trade-off between accuracy and computational cost.

We also present the confusion matrices of our proposed DocXClassifier-XL model on the two datasets in Fig. 4. As evident from Fig. 4a, majority of the classes were classified correctly to a large extent, but some of the classes were quite strongly confused with the others. For example, the two classes Presentation and Scientific Report showed an overlap of 3-4%. This finding is similar to that reported by Kanchi *et al.* (2022) [38, Fig. 9] on their multimodal approach. In contrast to their results, however, our approach performed better in distinguishing between Scientific Report and Scientific Publication classes. Overall, our approach fell short of their multimodal approach mainly on the Form, Questionnaire, and Scientific Report classes, suggesting that these three classes must benefit strongly from textual features of the documents.

Results on Tobacco3482. On the Tobacco3482 dataset, we observed a similar behavior to RVL-CDIP [17], where the DocXClassifier-L with RVL-CDIP [17] pre-training improved the classification accuracy by more than 1.53% over

TABLE 3. Ablation study of different training and data augmentation strategies.

Model	Accuracy (RVL-CDIP)	# of Parameters
ConvNeXt-B/224 (Aug_{Basic})	92.10%	87.6M
ConvNeXt-B/224 (Aug_{Basic} + $Aug_{cutmixup}$)	92.63%	87.6M
ConvNeXt-B-384 (Aug_{Basic})	93.13%	87.6M
ConvNeXt-B/384 (Aug_{Basic} + $Aug_{cutmixup}$)	93.60%	87.6M
ConvNeXt-B/384 ($Aug_{RandAug+Erase}$)	93.21%	87.6M
ConvNeXt-B/384 ($Aug_{RandAug+Erase}$ + $Aug_{cutmixup}$)	93.74%	87.6M
ConvNeXt-L/384 ($Aug_{RandAug+Erase}$ + $Aug_{cutmixup}$)	93.75%	196M
ConvNeXt-XL-384 ($Aug_{RandAug+Erase}$ + $Aug_{cutmixup}$)	93.81%	348M
ConvNeXt-B/384 ($Aug_{RandAug+Erase}$ + $Aug_{cutmixup}$ + EMA)	94.04%	87.6M
ConvNeXt-L/384 ($Aug_{RandAug+Erase}$ + $Aug_{cutmixup}$ + EMA)	94.15%	196M
ConvNeXt-XL/384 ($Aug_{RandAug+Erase}$ + $Aug_{cutmixup}$ + EMA)	94.17%	348M
DocXClassifier-B/384 ($Aug_{RandAug+Erase}$ + $Aug_{cutmixup}$ + EMA)	**94.00%**	95.4M
DocXClassifier-L/384 ($Aug_{RandAug+Erase}$ + $Aug_{cutmixup}$ + EMA)	**94.15%**	204M
DocXClassifier-XL/384 ($Aug_{RandAug+Erase}$ + $Aug_{cutmixup}$ + EMA)	**94.17%**	356M

the previous state-of-the-art approach for image-based classification whereas our lightest model DocXClassifier-B presented a 1.25% increase. Additionally, all of our proposed models even performed better than the two-stream combination of EfficientNet and BERT proposed by Ferrando *et al.* (2020) [1]. With only ImageNet pre-training, we achieved an accuracy of 90.14% on the Tobacco3482 dataset, which is not only the highest reported image-based classification accuracy, but also comparable to the recently presented multimodal approach [38] based on the combination of EfficientNet and Hierarchical Attention Networks, which achieved an accuracy of 90.3%.

We also present the confusion matrix for DocXClassifier-XL with RVL-CDIP [17] pre-training on the Tobacco3482 dataset, as shown in Fig. 4b. Similar to the case of RVL-CDIP, the majority of the classes showed a good behavior but only few classes were highly misclassified. For instance, the Scientific class was mainly confused with the Report class, which can be explained by the fact that these classes typically exhibit similar visual semantics. These findings are again very similar to the results of Kanchi *et al.* (2022) [38, Fig. 10] who found a large overlap between the Scientific and Report classes. On the other hand, our approach performed better on the ADVE class than their multimodal approach. This suggests that our visual representations are much richer than the EfficientNet network, since the classification of ADVE class in general depends largely on visual content.

Runtime Evaluation. In this section, we assess the runtime performance of our proposed models. It is common for document classification models to be used in fast-paced real-time scenarios, and therefore the runtime performance of these models is an important consideration. We evaluated the runtime performance of our models in terms of throughput on a single A100 GPU with a batch size of 256. The results are shown in the Table. 2. As can be seen, the average inference time per image for the DocXClassifier-B, DocXClassifier-L, and DocXClassifier-XL models was 6.5 ms, 10 ms, and 16 ms, respectively. In contrast, the throughput for each model was 153 frames/s, 100 frames/s, and 62 frames/s, respectively. It can be observed that the inference times of our proposed models are slightly higher compared to previous models such as ResNet-50, VGG-16, and EfficientNet-B4. However, we consider this to be a minor trade-off for the improved performance and real-time interpretability maps. Overall, we can conclude that the proposed models are highly suitable for real-time applications. Note that we did not find any difference in throughput performance between the DocXClassifier models and the corresponding ConvNext models. This makes sense since the number of parameters added by the attention-based pooling layer in the DocXclassifier models was relatively insignificant compared to the actual sizes of the models.

D. ABLATION STUDY

In this section, we present the results of our ablation study, in which we experimented with different sets of configurations to analyze the effects of data augmentation and preprocessing techniques on model performance. The results of the study are summarized in the Table 3. Looking for the best strategy for data augmentation, and training, we started with the base ConvNeXt-B network, a standard input resolution of 224x224, and a simple pre-processing scheme, referred to as Aug_{Basic}, which involved only downscaling the images to the network resolution, converting the images from grayscale to RGB, and then applying ImageNet normalization. Such a pre-processing scheme has been widely used in the past [17], [47] and therefore provides a good comparison.

As can be seen in the table, despite all the architectural improvements, the ConvNeXt model did not perform particularly well with this scheme, achieving only 92.10% accuracy which is comparable to the previous works [71]. Adding CutMix [41] and Mixup [32] data during training, denoted by $Aug_{cutmixup}$, resulted in a significant increase in network performance from 92.10% to 92.63%. Next, we changed the resolution of the network from 224x224 to 384x384 as previously done [1] and trained the network both with and without $Aug_{cutmixup}$. It is evident that increasing the resolution had a very significant impact on performance. The accuracy increased from 92.10% to 93.13% with Aug_{basic} and from 92.63% to 93.60% with $Aug_{cutmixup}$. Then, we replaced the Aug_{Basic} augmentation with a combination of RandAugment [31] and RandomErasing [33], which we refer to as $Aug_{RandAug+Erase}$. With this replacement, we again trained the network with and without $Aug_{cutmixup}$ and report the accuracy. As shown, using $Aug_{RandAug+Erase}$ slightly improved the performance of the network, from 93.13% to 93.21% and from 93.60% to 93.74% with and without $Aug_{cutmixup}$ during training, respectively. Additionally, we trained the ConvNeXt-L and ConvNeXt-XL networks with this final configuration and report their accuracy. It can be observed that ConvNeXt-L showed no significant improvement over ConvNeXt-B, possibly due to overfitting.

As mentioned in Sec. IV-B, we additionally computed the accuracy with and without EMA for all three variants. As shown, models with EMA performed significantly better with accuracies of 94.04%, 94.15% and 94.17% than the base models with accuracies of 93.74%, 93.75% and 93.81% respectively. Finally, we replaced the global average pooling

TABLE 4. Clean accuracy, clean error, mCE, and the corruption error (CE) values across different distortion types for each model are listed. AlexNet (Unnormalized) shows the actual magnitude of the errors caused by the distortions.

Model	Acc_{clean} (%)		E_{clean} (%)		mCE (%)		Rel. mCE (%)		Rel. $mCE_{1\%}$ (%)		Noise (% CE)								Digital (% CE)							
											Gaussian (%)		Shot (%)		Fibrous (%)		Multiscale (%)		Brightness (%)		Contrast (%)		Pixelate (%)		JPEG (%)	
	R	T	R	T	R	T	R	T	R	T	R	T	R	T	R	T	R	T	R	T	R	T	R	T	R	T
AlexNet (Unnormalized)	87.9	88.7	12.1	11.3	21.7	22.0	-	-	-	-	16	12	16	12	29	27	45	45	19	18	37	45	12	11	12	12
AlexNet [36]	87.9	88.7	12.1	11.3	100.0	100.0	100.0	100.0	100.0	100.0	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
VGG-16 [36]	91.0	93.7	9.0	6.3	65.3	49.0	43.6	37.1	39.8	31.0	59	55	59	58	54	45	55	43	61	42	59	30	73	56	74	54
GoogleNet [36]	90.5	90.9	9.5	9.1	85.1	96.0	100.5	329.8	88.2	160.2	88	127	88	127	131	167	122	135	65	77	73	78	80	98	83	103
ResNet-50 [36]	90.4	92.0	9.6	8.0	93.9	105.0	121.2	467.4	107.8	225.4	99	140	99	138	179	268	161	187	63	75	104	107	81	94	87	103
EfficientNet-B1 [1]	92.7	94.4	7.3	5.6	63.9	63.0	70.5	162.8	61.9	87.6	56	64	56	62	95	156	102	131	**44**	38	37	38	61	59	63	58
EfficientNet-B4 [1]	92.6	93.7	7.4	6.3	56.4	51.5	54.8	47.8	44.6	32.5	54	56	54	58	62	76	64	90	45	38	**32**	30	63	56	63	60
ViT-B/16 [11]	87.1	89.3	12.9	10.7	101.5	104.3	112.7	323.7	92.5	132.4	106	106	108	105	101	137	90	95	88	86	63	64	108	118	109	114
ViT-L/32 [11]	85.9	86.9	14.1	13.1	105.4	99.6	98.0	78.6	79.1	51.9	105	117	105	110	116	114	126	118	99	88	73	56	115	114	117	123
DocXClassifier-B (Ours)	**94.0**	**95.3**	**6.0**	**4.7**	**41.2**	**29.3**	**24.1**	**8.1**	**20.8**	**6.8**	**37**	**41**	**37**	**42**	**24**	**24**	**17**	**12**	48	**28**	84	**14**	**50**	**37**	**51**	**35**
DocXClassifier-L (Ours)	**94.2**	**95.6**	**5.9**	**4.4**	**40.3**	**27.2**	**25.4**	**2.6**	**21.2**	**2.6**	**37**	**35**	**36**	**35**	**24**	**21**	**19**	**11**	48	**25**	78	**11**	**49**	**36**	**49**	**37**
DocXClassifier-XL (Ours)	**94.2**	**95.4**	**5.9**	**4.6**	**41.7**	**34.0**	**27.9**	87.9	**23.9**	37.9	**37**	**44**	**37**	**44**	**24**	**23**	**20**	**13**	53	**31**	95	**13**	**48**	**46**	**49**	**47**

Model	Blur (% CE)												Geometric Distortions (% CE)						Documents Specific Distortions (% CE)							
	Defocus (%)		Motion (%)		Zoom (%)		Binary (%)		Gaussian(%)		Noisy Binary(%)		Affine (%)		Scale (%)		Elastic (%)		Surf Dist. (%)		Rand Dist. (%)		Blotches (%)		Threshold (%)	
	R	T	R	T	R	T	R	T	R	T	R	T	R	T	R	T	R	T	R	T	R	T	R	T	R	T
AlexNet (Unnormalized)	17	23	24	34	21	26	15	11	15	18	29	15	25	23	25	22	16	17	12	11	12	12	23	21	15	11
AlexNet [36]	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
VGG-16 [36]	57	31	44	23	53	35	75	59	63	37	65	53	84	72	80	64	69	50	74	61	74	55	60	44	77	63
GoogleNet [36]	88	98	84	90	95	93	91	74	86	98	79	76	74	68	76	78	76	93	80	99	80	93	60	55	87	90
ResNet-50 [36]	93	94	83	85	106	112	94	70	89	94	82	79	76	69	77	72	81	85	80	91	81	93	67	57	90	89
EfficientNet-B1 [1]	64	50	98	79	91	70	74	55	63	50	50	57	51	49	58	51	56	49	60	59	61	53	39	38	62	56
EfficientNet-B4 [1]	54	42	56	41	70	48	72	56	56	40	50	52	50	44	58	46	56	44	61	58	62	54	40	33	61	61
ViT-B/16 [11]	102	107	79	81	103	97	126	100	102	106	100	93	104	102	103	99	115	120	115	119	109	178	85	61	117	105
ViT-L/32 [11]	96	85	71	63	84	71	123	105	102	95	82	90	111	101	108	96	124	116	129	122	117	120	92	68	117	121
DocXClassifier-B (Ours)	**41**	**21**	**37**	**18**	**42**	**23**	**50**	**44**	**44**	**23**	**23**	**34**	**29**	**23**	**36**	**27**	**40**	**28**	**49**	**42**	**50**	**36**	**31**	**21**	**44**	**41**
DocXClassifier-L (Ours)	**43**	**18**	**35**	**16**	**41**	**23**	**48**	**38**	**44**	**23**	**22**	**29**	**28**	**23**	**35**	**28**	**39**	**26**	**48**	**38**	**49**	**36**	**29**	**23**	**43**	**38**
DocXClassifier-XL (Ours)	**42**	**24**	**37**	**20**	**43**	**31**	**49**	**51**	**43**	**30**	**22**	**36**	**28**	**27**	**34**	**29**	**39**	**34**	**48**	**49**	**49**	**45**	**38**	**27**	**42**	**51**

R: RVL-CDIP-D, T: Tobacco3482-D

and linear classification head of the original ConvNeXt models with the proposed attention-based pooling to construct the DocXClassifier models and re-trained them as described in Sec. IV-B. As can be seen from Table 3, on RVL-CDIP, we did not notice any change in accuracy when we switched the models from ConvNeXt to DocXClassifier. However, we did notice a slight improvement in performance on the Tobacco3482 dataset during our experiments. In addition to performance, it can also be seen that adding the attention-based pooling mechanism to the models only adds about $\sim$8M parameters to the model, which is insignificant compared to the overall size of the models.

E. EVALUATION OF MODEL ROBUSTNESS

In this section, we present a quantitative evaluation of the robustness of our proposed DocXClassifier models using the two benchmark datasets, RVL-CDIP-D and Tobacco3482-D [11], as discussed in Sec. IV-A. We use two standard robustness evaluation metrics, Mean Corruption Error (mCE) [10] and relative Mean Corruption Error (Rel. mCE) [10] to evaluate the robustness of the models and use AlexNet [75] as a baseline for computing these metrics as previously done in [11]. The mCE metric captures the overall decrease in model performance with the introduction of data corruptions, whereas Rel. mCE describes the decrease in model performance relative to its performance on the clean dataset. Both metrics are useful for assessing the robustness of the model against novel data distortions. For more details regarding these metrics, see Appendix A. In addition to Rel. mCE, we also propose an additional metric called Rel. $mCE_{1\%}$, in which we set the minimum possible baseline error to 1%. Since Rel. mCE is computed by dividing the relative CE of the model over the relative CE of the baseline, for some distortions where the baseline relative CE reaches close to zero (or even zero), Rel. mCE can produce highly exaggerated results that may not accurately represent the model's robustness. By setting a minimum baseline error of 1%, we ensure that division is not performed with extremely small numbers, resulting in more stable and interpretable values for the metric.

The results of this evaluation are presented in Table 4, where we list the errors on the clean datasets, the corruption error (CE) introduced by each individual distortion type, and finally, the mCE and Rel. mCEs across all distortion types. For the baseline AlexNet model, we also present the actual error rates introduced by each distortion type under *AlexNet (Unnormalized)* case, relative to which the CEs of all other models are computed. As evident from the table, our proposed DocXClassifier models significantly outperformed existing approaches in terms of robustness on both the RVL-CDIP-D and Tobacco3482-D datasets, achieving mCE values as low as $\sim$40% on the RVL-CDIP-D datasets and $\sim$22% on the Tobacco3482-D dataset, respectively. It is also evident from the table that our proposed models consistently demonstrated significantly superior robustness compared to other models across the majority (18 out of 21) of distortion types. Interestingly, however, we noticed the our proposed models were significantly affected by increasing image contrast on the RVL-CDIP-D dataset.

It can be further observed from the table that our proposed models also demonstrated significantly high relative robustness, achieving Rel. mCE and Rel. $mCE_{1\%}$ values as low as 24.1% and 20.8%, respectively, on the RVL-CDIP-D dataset. On Tobacco3482-D, both DocXClassifier-B and DocXClassifier-L models again demonstrated superior relative robustness, achieving Rel. mCE values as low as 8.1% and 2.6%, respectively. In contrast, however, the DocXClassifier-XL model performed considerably worse on this dataset. A possible explanation for this discrepancy could be model overfitting on the small-scale Tobacco3482 dataset

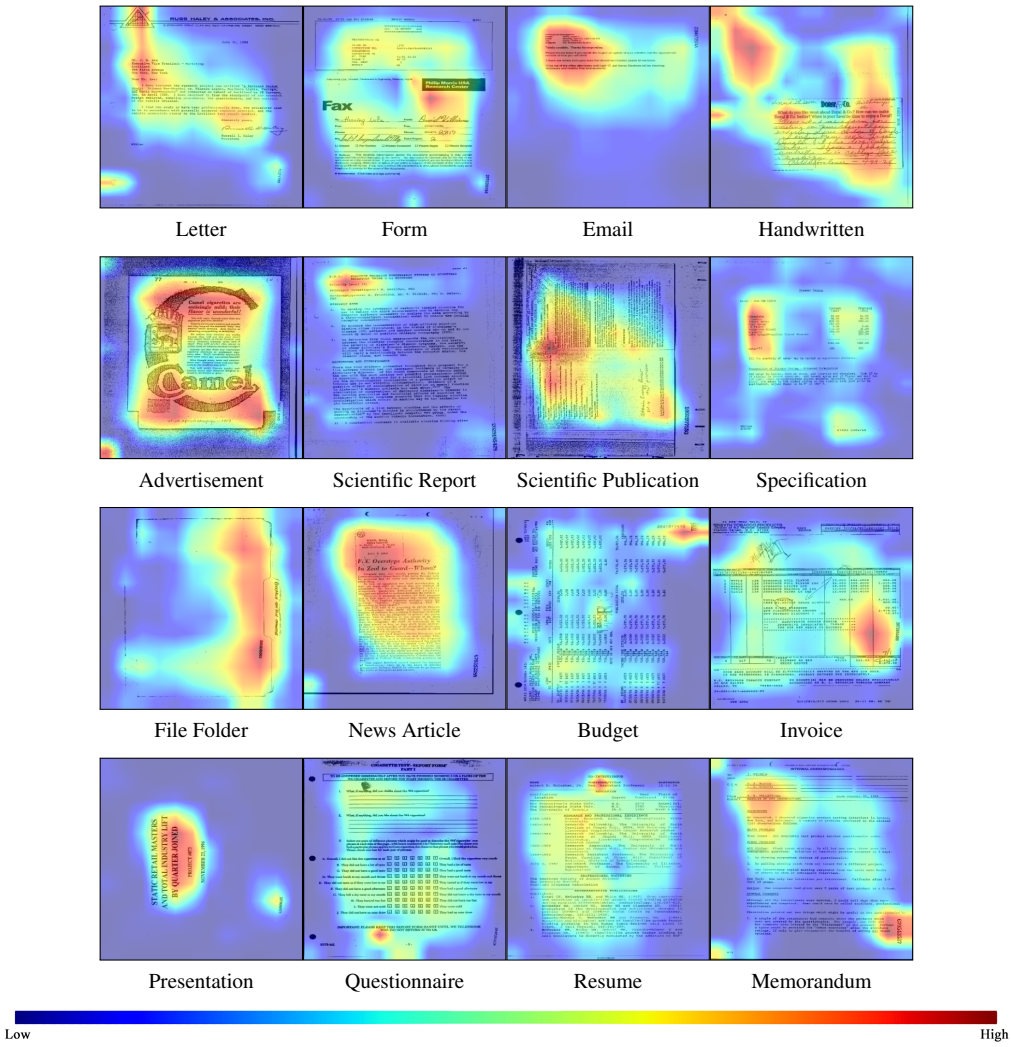


FIGURE 5. Attention maps generated using the DocXClassifier-B model for a few randomly selected samples from the RVL-CDIP dataset. The feature importance intensity ranges from from blue (low) to red (high).

due to its large number of parameters. However, it is also worth mentioning that on the Tobacco3482-D dataset, we also observed significant differences in Rel. mCE and Rel. $\text{mCE}_{1\%}$, primarily due to the extremely low relative baseline error rates. By examining the Rel. $\text{mCE}_{1\%}$ values in this case, it can be observed that overall, DocXclassifier-XL showed robustness on a similar scale to EfficientNet-B4 and VGG-16 models while demonstrating significantly superior performance on the clean dataset.

F. EVALUATION OF MODEL INTERPRETABILITY

In this section, we present a qualitative analysis of the interpretability of the proposed DocXClassifier models. As explained in Sec. III, we utilized an attention-based mechanism to perform a weighted aggregation of the image feature vectors, the result of which is then fed into a feed-forward network for classification. This weighted aggregation results in attention weights for each predicted output, which directly represent the importance the model assigns to each of the feature vectors (and thus regions or patches) of the image. We use these attention weights to generate the attention map for a predicted image and upsample it to the base image resolution for visualization.

The attention maps for a few randomly selected samples from the RVL-CDIP dataset, generated using the DocXClassifier-B variant, are visualized in Fig. 5. Here, we only visualize the positively predicted samples for each of the 16 classes. It can be observed that the model had learned to focus on specific regions of the image for each class. For instance, for the Email, Letter, and Memorandum classes, the network had learned to focus on the document header with author, recipients, and subject information. For other classes, the network focused on class-specific information, such as the resume title, qualifications and experience of the subject in the Resume class or items lists in the Specification class. One interesting observation was that, in several instances,

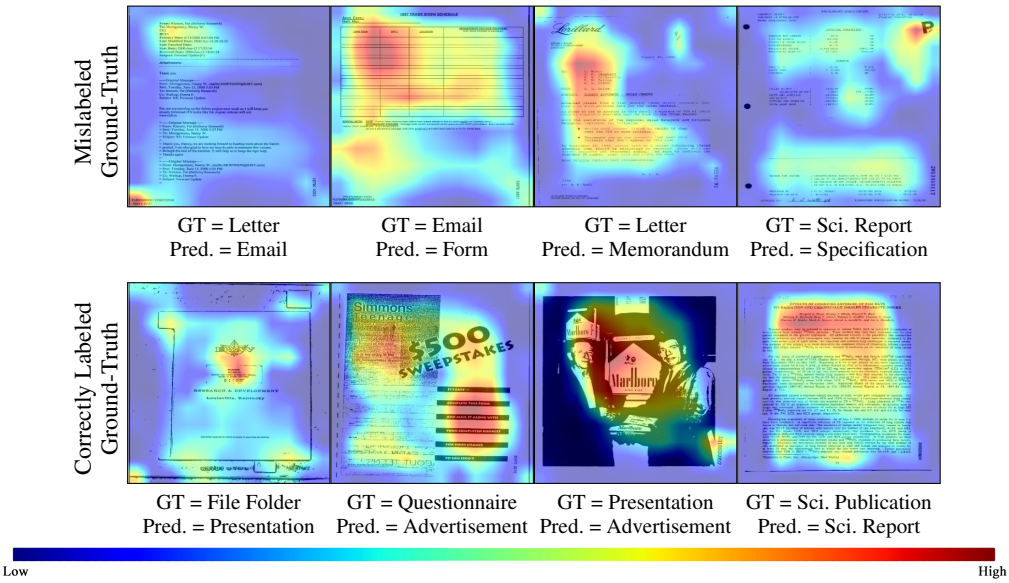


FIGURE 6. Attention maps generated using the DocXClassifier-B model for a selection of misclassified samples. Top row shows samples for which the ground-truth (GT) is mislabeled but the model predicts the correct class. Bottom row shows samples where the ground-truth (GT) is correctly assigned but the model fails to predict the correct class.

the model assigned significant importance to identification numbers, particularly within classes like Letter, Memorandum, Presentation, and Budget. This raises the question of whether these numbers hold some class-specific information within them; however, we will leave this investigation to future work.

We further analyze attention maps for samples where the network made false predictions. A notable observation in this scenario was that for many samples, the model actually made the correct predictions, but the ground-truth was falsely annotated. A few samples of this type are visualized in Fig. 6. In the top row, we display attention maps for instances where incorrect ground-truth labels were assigned in the dataset, yet the network correctly predicted the class. For instance, it can be observed from the top row that the model correctly predicted the Email, Form, Memorandum, and Specification classes for these samples, while their ground-truth labels were incorrectly assigned. By utilizing the attention maps, we can also analyze which regions were considered important by the model while making these predictions. The bottom row illustrates attention maps for samples with correctly assigned ground-truth labels on which the model failed to predict the correct class. It can be observed that the model failed to correctly predict the classes for samples that were difficult to distinguish between multiple types. For instance, both the 2nd and 3rd samples in the bottom row bear a strong resemblance to the advertisement samples in the RVL-CDIP dataset, as predicted by the model. However, they are categorized into different classes. Notably, the model appears to focus on specific advertisement-related imagery for making these predictions.

Finally, we conduct a qualitative comparison of our approach with existing feature-attribution-based methods. We compare our approach with five different state-of-the-art feature attribution-based approaches: Saliency [76], IntegratedGradients [57], DeepLIFT [58], LIME [20], KernelSHAP [22], and Occlusion [77]. For all these approaches, we use the implementations provided in the Captum library [2] and utilize standard settings to generate the feature importance maps. In addition, since DocXClassifier is designed as an enhancement to the base ConvNeXt model to introduce interpretability, we apply the feature-attribution techniques to the base ConvNeXt-based classifier to assess the performance of the upgrade in comparison. Note that both classifiers in this case use the same frozen weights of the base ConvNeXt feature backbone as explained in Section. IV-B, and perform equally well on the dataset. The results are present in Fig. 7 where we visualize the attribution maps generated using all the different approaches for four randomly selected samples from the RVL-CDIP dataset. A few interesting conclusions can be drawn from the results. First, it can be directly observed that our approach proved to be considerably more visually interpretable than most of the approaches. Interestingly, the importance maps generated by DocXClassifier model were slightly similar to those generated by the LIME [20] and KernelSHAP [22] approaches. However, both LIME [20] and KernelSHAP [22] were difficult to interpret due to the region segmentation. In comparison, our approach produced smoother importance maps over the different image regions. The gradient-based approaches, on the other hand, surprisingly failed to produce any reasonable results on this model resulting in noisy at-

[2]https://github.com/pytorch/captum

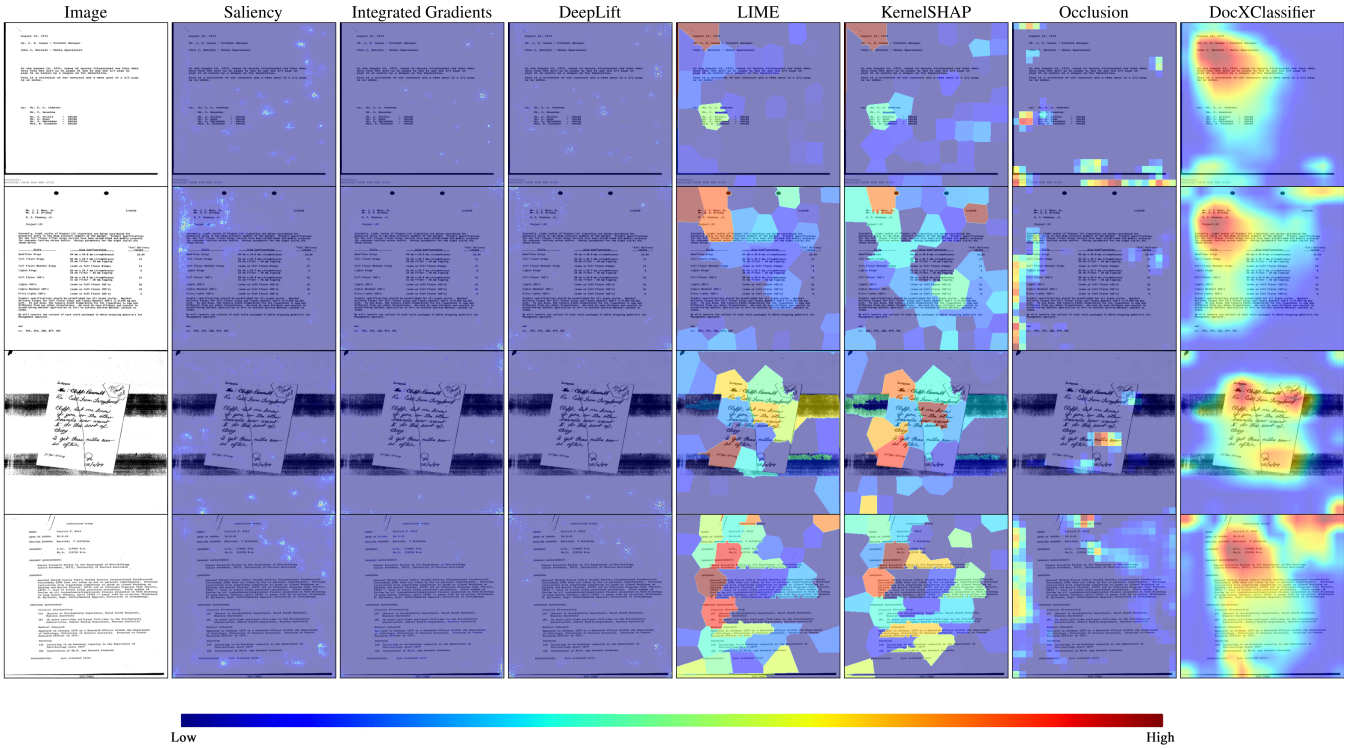


FIGURE 7. Comparison of the feature importance maps generated by DocXClassifier-B with existing feature-attribution-based XAI methods. We generate the feature attribution maps for existing techniques by applying these methods directly to the base ConvNeXt-B model.

tribution maps with little to no importance assigned to any region of the images. Similarly, while Occlusion [77] did assign importance to different image regions, its results were also difficult to interpret. Overall, we can conclude that our approach is well-suited for generating human-interpretable attribution maps, eliminating the need to resort to post-hoc interpretability approaches to generate the explanations.

V. CONCLUSION

Model interpretability and model robustness are two main challenges when it comes to safe and efficient deployment of deep neural networks in real-world scenarios. In this work, we addressed these challenges in the context of document image classification and introduced DocXClassifier, an inherently interpretable deep convolutional neural network which holds the capability to efficiently generate feature importance maps at test time. Furthermore, in order to enhance model robustness to out-of-distribution data, we presented a training strategy that incorporates advanced data augmentation strategies and training techniques which have been previously left unexplored in this domain. Through extensive evaluation, we demonstrated that our proposed training strategy significantly improves both performance and robustness, outperforming all existing image-based document classification approaches in both aspects while remaining runtime-efficient. Furthermore, we evaluated the interpretability of our approach in comparison to existing explainability methods and demonstrated its superiority in generating human-interpretable feature attribution maps. By tackling both robustness and interpretability challenges simultaneously, our work presents a significant step towards secure and robust deployment of deep neural networks for document image classification.

REFERENCES

[1] J. Ferrando, J. L. Domínguez, J. Torres, R. García, D. García, D. Garrido, J. Cortada, and M. Valero, "Improving accuracy and speeding up document image classification through parallel systems," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 12138 LNCS, pp. 387–400, 2020.

[2] N. Audebert, C. Herold, K. Slimani, and C. Vidal, "Multimodal deep networks for text and image-based document classification," in Commun. Comput. Inf. Sci., vol. 1167 CCIS, pp. 427–443, Springer, Cham, sep 2020.

[3] Y. Xu, Y. Xu, T. Lv, L. Cui, F. Wei, G. Wang, Y. Lu, D. Florencio, C. Zhang, W. Che, M. Zhang, and L. Zhou, "LayoutLMv2: Multi-modal Pre-training for Visually-rich Document Understanding," pp. 2579–2591, Association for Computational Linguistics (ACL), dec 2021.

[4] R. Powalski, Ł. Borchmann, D. Jurkiewicz, T. Dwojak, M. Pietruszka, and G. Pałka, "Going Full-TILT Boogie on Document Understanding with Text-Image-Layout Transformer," in Doc. Anal. Recognit. – ICDAR 2021, vol. 12822 LNCS, pp. 732–747, 2021.

[5] Z. Shen, R. Zhang, M. Dell, B. C. G. Lee, J. Carlson, and W. Li, "Layoutparser: A unified toolkit for deep learning based document image analysis," arXiv preprint arXiv:2103.15348, 2021.

[6] D. V. Carvalho, E. M. Pereira, and J. S. Cardoso, "Machine learning interpretability: A survey on methods and metrics," Electronics, vol. 8, no. 8, 2019.

[7] M. Honegger, "Shedding light on black box machine learning algorithms: Development of an axiomatic framework to assess the quality of methods that explain individual predictions," ArXiv, vol. abs/1808.05054, 2018.

[8] S. Dodge and L. Karam, "A study and comparison of human and deep learning recognition performance under visual distortions," 2017 26th Int. Conf. Comput. Commun. Networks, ICCCN 2017, 2017.

[9] B. Recht, R. Roelofs, L. Schmidt, and V. Shankar, "Do ImageNet classifiers generalize to ImageNet?," 36th Int. Conf. Mach. Learn. ICML 2019, vol. 2019-June, pp. 9413–9424, 2019.

[10] D. Hendrycks and T. Dietterich, "Benchmarking neural network robustness to common corruptions and perturbations," 7th Int. Conf. Learn. Represent. ICLR 2019, pp. 1–16, 2019.

[11] Saifullah, S. A. Siddiqui, S. Agne, A. Dengel, and S. Ahmed, "Are deep models robust against real distortions? a case study on document image classification," in 2022 26th International Conference on Pattern Recognition (ICPR), pp. 1628–1635, 2022.

[12] E. Ntoutsi, P. Fafalios, U. Gadiraju, V. Iosifidis, W. Nejdl, M.-E. Vidal, S. Ruggieri, F. Turini, S. Papadopoulos, E. Krasanakis, I. Kompatsiaris, K. Kinder-Kurlanda, C. Wagner, F. Karimi, M. Fernandez, H. Alani, B. Berendt, T. Kruegel, C. Heinze, K. Broelemann, G. Kasneci, T. Tiropanis, and S. Staab, "Bias in data-driven artificial intelligence systems—an introductory survey," WIREs Data Mining and Knowledge Discovery, vol. 10, no. 3, p. e1356, 2020.

[13] A. Lucieri, F. Schmeisser, C. P. Balada, S. A. Siddiqui, A. Dengel, and S. Ahmed, "Revisiting the shape-bias of deep learning for dermoscopic skin lesion classification," in Medical Image Understanding and Analysis (G. Yang, A. Aviles-Rivero, M. Roberts, and C.-B. Schönlieb, eds.), (Cham), pp. 46–61, Springer International Publishing, 2022.

[14] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel, "Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness," 2022.

[15] H. Hosseini, B. Xiao, and R. Poovendran, "Google's cloud vision API is not robust to noise," Proc. - 16th IEEE Int. Conf. Mach. Learn. Appl. ICMLA 2017, vol. 2017-December, pp. 101–105, 2017.

[16] R. Geirhos, D. H. J. Janssen, H. H. Schütt, J. Rauber, M. Bethge, and F. A. Wichmann, "Comparing deep neural networks against humans: object recognition when the signal gets weaker," 2017.

[17] A. W. Harley, A. Ufkes, and K. G. Derpanis, "Evaluation of deep convolutional nets for document image classification and retrieval," Proc. Int. Conf. Doc. Anal. Recognition, ICDAR, vol. 2015-Novem, pp. 991–995, 2015.

[18] A. Groleau, K. W. Chee, S. Larson, S. Maini, and J. Boarman, "Augraphy: A data augmentation library for document images," 2023.

[19] R. D. Lins, R. B. Bernardino, R. d. S. Barboza, and S. J. Simske, "Binarization of photographed documents image quality, processing time and size assessment," in Proceedings of the 22nd ACM Symposium on Document Engineering, pp. 1–10, 2022.

[20] M. Ribeiro, S. Singh, and C. Guestrin, ""why should i trust you?": Explaining the predictions of any classifier," pp. 97–101, 02 2016.

[21] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," International Journal of Computer Vision, vol. 128, p. 336–359, Oct 2019.

[22] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS'17, (Red Hook, NY, USA), p. 4768–4777, Curran Associates Inc., 2017.

[23] O. Lang, Y. Gandelsman, M. Yarom, Y. Wald, G. Elidan, A. Hassidim, W. T. Freeman, P. Isola, A. Globerson, M. Irani, and I. Mosseri, "Explaining in style: Training a gan to explain a classifier in stylespace," ArXiv, vol. abs/2104.13369, 2021.

[24] D. Nemirovsky, N. Thiebaut, Y. Xu, and A. Gupta, "Countergan: Generating realistic counterfactuals with residual generative adversarial nets," arXiv preprint arXiv:2009.05199, 2020.

[25] N. Xie, G. Ras, M. van Gerven, and D. Doran, "Explainable deep learning: A field guide for the uninitiated," 04 2020.

[26] M. Ivanovs, R. Kadikis, and K. Ozols, "Perturbation-based methods for explaining deep neural networks: A survey," Pattern Recognition Letters, vol. 150, pp. 228–234, 2021.

[27] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead," Nature machine intelligence, vol. 1, no. 5, pp. 206–215, 2019.

[28] Z. C. Lipton, "The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery.," Queue, vol. 16, p. 31–57, jun 2018.

[29] S. Zheng, Y. Song, T. Leung, and I. Goodfellow, "Improving the robustness of deep neural networks via stability training," 2016.

[30] D. Hendrycks, K. Zhao, S. Basart, J. Steinhardt, and D. Song, "Natural Adversarial Examples," 2019.

[31] E. Cubuk, B. Zoph, J. Shlens, and Q. Le, "Randaugment: Practical automated data augmentation with a reduced search space," pp. 3008–3017, 06 2020.

[32] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in International Conference on Learning Representations, 2018.

[33] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random erasing data augmentation," 2017.

[34] D. Hendrycks, N. Mu, E. D. Cubuk, B. Zoph, J. Gilmer, and B. Lakshminarayanan, "Augmix: A simple data processing method to improve robustness and uncertainty," 2020.

[35] N. Chen and D. Blostein, "A survey of document image classification: Problem statement, classifier architecture and performance evaluation," Int. J. Doc. Anal. Recognit., vol. 10, no. 1, pp. 1–16, 2007.

[36] M. Z. Afzal, A. Kolsch, S. Ahmed, and M. Liwicki, "Cutting the Error by Half: Investigation of Very Deep CNN and Advanced Training Strategies for Document Image Classification," Proc. Int. Conf. Doc. Anal. Recognition, ICDAR, vol. 1, pp. 883–888, 2017.

[37] M. N. Asim, M. U. G. Khan, M. I. Malik, K. Razzaque, A. Dengel, and S. Ahmed, "Two stream deep network for document image classification," Proc. Int. Conf. Doc. Anal. Recognition, ICDAR, pp. 1410–1416, 2019.

[38] S. Kanchi, A. Pagani, H. Mokayed, M. Liwicki, D. Stricker, and M. Z. Afzal, "Emmdocclassifier: Efficient multimodal document image classifier for scarce data," Applied Sciences, vol. 12, p. 1457, 01 2022.

[39] C. Tensmeyer and T. Martinez, "Analysis of Convolutional Neural Networks for Document Image Classification," Proc. Int. Conf. Doc. Anal. Recognition, ICDAR, vol. 1, pp. 388–393, 2017.

[40] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A convnet for the 2020s," 2022.

[41] S. Yun, D. Han, S. Chun, S. J. Oh, Y. Yoo, and J. Choe, "Cutmix: Regularization strategy to train strong classifiers with localizable features," pp. 6022–6031, 10 2019.

[42] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," 06 2016.

[43] S. Chen, Y. He, J. Sun, and S. Naoi, "Structured document classification by matching local salient features," Proc. - Int. Conf. Pattern Recognit., no. Icpr, pp. 653–656, 2012.

[44] J. Kumar, P. Ye, and D. Doermann, "Structural similarity for document image classification and retrieval," Pattern Recognit. Lett., vol. 43, no. 1, pp. 119–126, 2014.

[45] S. Baldi, S. Marinai, and G. Soda, "Using tree-grammars for training set expansion in page classification," Proc. Int. Conf. Doc. Anal. Recognition, ICDAR, vol. 2003-Janua, no. Icdar, pp. 829–833, 2003.

[46] L. Kang, J. Kumar, P. Ye, Y. Li, and D. Doermann, "Convolutional neural networks for document image classification," Proc. - Int. Conf. Pattern Recognit., pp. 3168–3172, 2014.

[47] 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," Proc. Int. Conf. Doc. Anal. Recognition, ICDAR, vol. 2015-Novem, pp. 1111–1115, 2015.

[48] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255, 2009.

[49] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in Proceedings of the 36th International Conference on Machine Learning (K. Chaudhuri and R. Salakhutdinov, eds.), vol. 97 of Proceedings of Machine Learning Research, pp. 6105–6114, PMLR, 09–15 Jun 2019.

[50] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," 2021.

[51] S. Sevim, S. İ. Omurca, and E. Ekinci, "Document image classification with vision transformers," in Electrical and Computer Engineering (M. N. Seyman, ed.), (Cham), pp. 68–81, Springer International Publishing, 2022.

[52] S. A. Siddiqui, A. Dengel, and S. Ahmed, "Analyzing the potential of zero-shot recognition for document image classification," in Document Analysis and Recognition – ICDAR 2021 (J. Lladós, D. Lopresti, and S. Uchida, eds.), (Cham), pp. 293–304, Springer International Publishing, 2021.

[53] T. Dauphinee, N. Patel, and M. Rashidi, "Modular multimodal architecture for document classification," 2019.

[54] Y. Xiong, Z. Dai, Y. Liu, and X. Ding, "Document image classification method based on graph convolutional network," in Neural Information

Processing (T. Mantoro, M. Lee, M. A. Ayu, K. W. Wong, and A. N. Hidayanto, eds.), (Cham), pp. 317–329, Springer International Publishing, 2021.
[55] Y. Xu, M. Li, L. Cui, S. Huang, F. Wei, and M. Zhou, "LayoutLM: Pre-training of Text and Layout for Document Image Understanding," Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min., vol. 20, pp. 1192–1200, 2020.
[56] Z. Tang, Z. Yang, G. Wang, Y. Fang, Y. Liu, C. Zhu, M. Zeng, C. Zhang, and M. Bansal, "Unifying vision, text, and layout for universal document processing," 2023.
[57] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," in International conference on machine learning, pp. 3319–3328, PMLR, 2017.
[58] A. Shrikumar, P. Greenside, and A. Kundaje, "Learning important features through propagating activation differences," in Proceedings of the 34th International Conference on Machine Learning - Volume 70, ICML'17, p. 3145–3153, JMLR.org, 2017.
[59] J. Adebayo, J. Gilmer, M. Muelly, I. Goodfellow, M. Hardt, and B. Kim, "Sanity checks for saliency maps," 2020.
[60] C. Chen, O. Li, D. Tao, A. Barnett, C. Rudin, and J. K. Su, "This looks like that: Deep learning for interpretable image recognition," in Advances in Neural Information Processing Systems (H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, eds.), vol. 32, Curran Associates, Inc., 2019.
[61] P. Angelov and E. Soares, "Towards explainable deep neural networks (xdnn)," 2019.
[62] H. Touvron, M. Cord, A. El-Nouby, P. Bojanowski, A. Joulin, G. Synnaeve, and H. Jégou, "Augmenting convolutional networks with attention-based aggregation," 2021.
[63] B. Kim, M. Wattenberg, J. Gilmer, C. Cai, J. Wexler, F. Viegas, and R. Sayres, "Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav)," 2018.
[64] X. Yuan, P. He, Q. Zhu, and X. Li, "Adversarial examples: Attacks and defenses for deep learning," IEEE Transactions on Neural Networks and Learning Systems, vol. 30, no. 9, pp. 2805–2824, 2019.
[65] E. D. Cubuk, B. Zoph, D. Mane, V. Vasudevan, and Q. V. Le, "Autoaugment: Learning augmentation policies from data," 2019.
[66] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778, 2016.
[67] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," 2021.
[68] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," 2017.
[69] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2017.
[70] G. Huang, Y. Sun, Z. Liu, D. Sedra, and K. Q. Weinberger, "Deep networks with stochastic depth," in Computer Vision – ECCV 2016 (B. Leibe, J. Matas, N. Sebe, and M. Welling, eds.), (Cham), pp. 646–661, Springer International Publishing, 2016.
[71] A. Das, S. Roy, U. Bhattacharya, and S. K. Parui, "Document Image Classification with Intra-Domain Transfer Learning and Stacked Generalization of Deep Convolutional Neural Networks," Proc. - Int. Conf. Pattern Recognit., vol. 2018-Augus, pp. 3180–3185, 2018.
[72] R. Sarkhel and A. Nandi, "Deterministic routing between layout abstractions for multi-scale classification of visually rich documents," pp. 3360–3366, 08 2019.
[73] T. Dauphinee, N. Patel, and M. Rashidi, "Modular multimodal architecture for document classification," 12 2019.
[74] S. Bakkali, Z. Ming, M. Coustaty, and M. Rusinol, "Visual and textual deep feature fusion for document image classification," pp. 2394–2403, 06 2020.
[75] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in Advances in Neural Information Processing Systems (F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, eds.), vol. 25, Curran Associates, Inc., 2012.
[76] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," arXiv preprint arXiv:1312.6034, 2013.
[77] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," 2013.

.

APPENDIX A ROBUSTNESS EVALUATION METRICS

The metrics Mean Corruption Error (mCE) [10] and relative Mean Corruption Error (Rel. mCE) [10] were both proposed by Hendrycks *et al.* (2019) [10] to evaluate the robustness of deep neural networks on distorted datasets. The robustness is generally computed relative to a baseline model, which we choose to be the AlexNet [75] model in this work, similar to what has been previosuly done by Saifullah *et al.* (2022) [11].

Mean Corruption Error (mCE). Let $E^{f}_{s,d}$ be the error rate of a trained classifier f on data corrupted by distortion type d with severity s, then the mean corruption error mCE of classifier f is defined as the total classification error of f with respect to the baseline classifier on the distorted dataset and can be calculated as follows:

$$\text{mCE}^{f} = \frac{1}{n_d}\sum_{d=1}^{n_d}\left[(\sum_{s=1}^{n_{s,d}} E^{f}_{s,d})/(\sum_{s=1}^{n_{s,d}} E^{f'}_{s,d})\right] \tag{2}$$

Where f' represents the baseline classifier used to normalize the distortion errors, n_d denotes the total number of distortion types applied to the data, and $n_{s,d}$ denotes the number of severity levels defined for each distortion.

Relative Mean Corruption Error (Rel. mCE). The second evaluation metric, namely relative mCE, computes the relative decline in the performance of a given classifier f with respect to its performance on the clean dataset and can be obtained by the following equation:

$$\text{Rel. mCE}^{f} = \frac{1}{n_d}\sum_{d=1}^{n_d}\left[(\sum_{s=1}^{n_{s,d}} E^{f}_{s,d} - E^{f}_{clean})/(\sum_{s=1}^{n_{s,d}} E^{f'}_{s,d} - E^{f'}_{clean})\right] \tag{3}$$

SAIFULLAH received the B.S. degree in mechanical engineering and the M.S. degree in robotics and intelligent machine engineering from the National University of Sciences and Technology (NUST), Pakistan. He is currently pursuing his Ph.D. at the University of Kaiserslautern and is working as a researcher at the German Research Center for Artificial Intelligence (DFKI GmbH) under the supervision of Prof. Dr. Prof. H. C. Andreas Dengel. His research interests include document understanding and analysis, explainability and robustness of deep learning models, and privacy preservation in deep learning.

STEFAN AGNE received the Diploma degree in computer science and the Ph.D. degree under the supervision of Dr. A. Dengel from the University of Kaiserslautern, Germany, in 1995, and the Ph.D. degree from the University of Bern, in 2008, under the supervision of Dr. H. Bunke. Since 1992, he has been working in document analysis and understanding with the German Research Center for Artificial Intelligence (DFKI GmbH), Germany. He is currently leading the topic area pattern recognition with the Smart Data and Knowledge Services Department, German Research Center for Artificial Intelligence, Kaiserslautern.

ANDREAS DENGEL received the Diploma degree in C.S. from the University of Kaiserslautern and the Ph.D. degree from the University of Stuttgart. He is the Scientific Director at the German Research Center for Artificial Intelligence (DFKI GmbH), Kaiserslautern. In 1993, he became a Professor at the Computer Science Department, University of Kaiserslautern, where he is currently the Chair of Knowledge-Based Systems, and since 2009, he has been a Professor (Kyakuin) at the Department of Computer Science and Information Systems, Osaka Prefecture University. He also worked at IBM, Siemens, and Xerox Parc. He is an author of more than 300 peer-reviewed scientific publications and supervised more than 170 Ph.D. and master's theses. His main scientific emphasis is in the areas of pattern recognition, document understanding, information retrieval, multimedia mining, semantic technologies, and social media. He is a member of several international advisory boards, chaired major international conferences, and founded several successful start-up companies. He is an IAPR Fellow and received prominent international awards. Moreover, he is a co-editor of international computer science journals and has written or edited 12 books.

SHERAZ AHMED is Senior Researcher at DFKI GmbH in Kaiserslautern, where he is leading the area of Time Series Analysis and Life Science. He received his MS and PhD degrees in Computer Science from TUK, Germany under the supervision of Prof. Dr. Prof. h.c. Andreas Dengel and Prof. Dr. habil. Marcus Liwicki. His PhD topic is Generic Methods for Information Segmentation in Document Images. Over the last few years, he has primarily worked on development of various systems for information segmentation in document images. His research interests include document understanding, generic segmentation framework for documents, pattern recognition, anomaly detection, Gene analysis, medical image analysis, and natural language processing. He has more than 100 publications on the said and related topics including three journal papers and two book chapters. He is a frequent reviewer of various journals and conferences including Pattern Recognition Letters, Neural Computing and Applications, IJDAR, ICDAR, ICFHR, and DAS.