Applied Intelligence (2023) 53:15295–15310
https://doi.org/10.1007/s10489-022-04306-5

A document image classification system fusing deep and machine learning models

Sevinç İlhan Omurca[1] · Ekin Ekinci[2] · Semih Sevim[3] · Eren Berk Edinç[1] · Süleyman Eken[4] · Ahmet Sayar[1]

Accepted: 27 October 2022 / Published online: 15 November 2022

Abstract

Artificial Intelligence (AI) technologies are now widely employed to overcome human-induced faults in a variety of systems used in our daily lives, thanks to the digital transformation.One example of such systems is online document tracking systems (DTS). The DTS's reliability and preferability are enhanced by automatic document classification and understanding features. Although automatic document classification systems can assist humans in document understanding tasks, most of of them are not designed to function with Portable Document Format (PDF), which contains text, tables or figures. In this study, we investigate separate ways to efficiently classify student documents that are uploaded in PDF format and are required for university education. We propose three possible techniques for this issue. The first approach is based on Optical Character Recognition (OCR) and traditional machine learning methods. The second is purely on deep learning. The third one is based on fusion of deep learning methods based on entropy. The proposed techniques can classify twelve distinct types of digital documents. The validity of the proposed methods has been verified by student affairs department of Kocaeli University in Turkey. The system has not only increased the efficiency of online document uploading steps for students, but also reduced the human cost for tracking the documents. The highest F-score (94.45%) is obtained by the ensemble of EfficientNetB3 and ExtraTree.

Keywords Document image classification · Document understanding · Deep learning · Machine learning · Ensemble learning

1 Introduction

Most of today's institutions and organizations choose to digitize their processes due to the digital world and its implications [1]. The digital creation of documents or transfer of non-digital documents to digital media is the first of these [2, 3]. Digitally created documents, such as in Extensible Markup Language (XML) or PDF formats, are machine readable, so it is easy to extract information from them. Document images, on the other hand, are manually made documents that are transmitted to digital media by means of a camera, printer, or other means; information cannot be extracted directly from such documents. While document images contain text in natural language, they can also be formatted as plain text, multi column, and various tables/forms/shapes. OCR has been widely used to extract text from document images [4], however, the visual structures of digital documents can also be descriptive. Document understanding is a challenging field because it requires combining the knowledge extracted from document visual structures and document textual features [5]. However, more accurate analysis results can be achieved in this way.

Understanding or analyzing documents is a time consuming and complex process for many public and private organizations. To overcome these manual and inefficient processes, AI methods have been used successfully to automatically classify document images or extract information from them. Among these methods, current and successful models are mostly built on deep neural network architectures both from a computer vision or Natural Language Processing (NLP) perspective, or a combination of them [6–10].

When large and corporate organizations are considered, universities with a huge number of students and many distinct types of document flow are among them. In this

✉ Ekin Ekinci
ekinekinci@subu.edu.tr

Extended author information available on the last page of the article.

paper, we aim to classify digital documents uploaded to the Kocaeli University digital document management system automatically. A digital document classification system can reduce routine work tasks, improve work efficiency, and resolve the contradictions caused by the rising number of incorrectly uploaded documents shortages. A digital document classification system like this can also help with intelligent document storage, management, and classification.

Students perform many distinct types of document uploads using document management systems provided to them throughout their university education. These documents can be in PDF format or scanned PDF document or an image. In the online document upload system of Kocaeli University's student affairs department, it was observed that the types of documents uploaded by students may be different from the expected document types. For example, several incorrect instances have been identified, such as uploading a student certificate instead of a transcript or a transfer paperwork instead of an identity card. Considering a large number of students in a university and the variety of document types, uploading wrong documents becomes critical for the correct follow-up of students' educational processes. These types of scenarios eventually lead to improper management of student processes such as adjustment and graduation, duplication of work, and greater labor expenses. In this paper, we aim to make the student affairs system more efficient and automated by applying machine learning and deep learning models to the field of digital document classification systems.

Our main contributions of this paper are as follows:

- By using machine learning and deep learning methods, we aim to classify digitized student documents.
- To find an accurate model, multiple algorithms have been explored in detail and compared using a real-world our own dataset.
- First, we learn textual features from content extracted by OCR and these features are fed into machine learning algorithms. Second, image-based features are extracted with CNN and used as input for CNN-based models. Finally, we introduce an ensemble document image classification that fuses content-based and image-based models based on the weighted mean.
- We show that the weighted-mean approach is effective for document image classification.

The remainder of this paper is organized as follows. In Section 2, we present related work on the document images classification. We discuss the details of the proposed work in Section 3. Section 4 firstly gives our own created dataset and secondly performance results, and then discussion. Finally, the last section presents the Section 5 of this work.

2 Related works

Document image classification has a long history in the field of document analysis and recognition, and its importance is growing by the day as the world digitizes. Textual data embedded in pixel values in document images should be extracted in order to achieve organizing, analyzing, retrieving, and efficient indexing. These tasks need classification of document images.

Many different approaches have been proposed for the classification of document images. However, these approaches are generally divided into three sections: (1) structure/layout based, (2) content based and (3) hybrid of previous approaches. While visual features are used for structure/layout-based classification OCRed version of document images are employed for content-based approaches.

We give an overview of cutting-edge works in the relevant topic in this area, which refer to one of the structure/layout-based, content-based, or hybrid document image classification methods.

Kumar et al. [11] built a codebook of SURF descriptors extracted from document images. These features were used to train random forest classifiers. The main disadvantage of this study was the proposed method could only classify structurally similar document images. The CNN model was used for the first time by Kang et al. [12] to classify document images. Although their model was simple and shallow, the model achieved superiority over structural similarity-based methods. Afzal et al. [13] used deep convolutional neural networks (DCNN) trained on ImageNet to classify document images into ten categories: news, letter, ad, note, scientific, form, email, memo, resume, and report. In their study, it was claimed that features extracted from the structure of the document were the most informative features in document image classification. Harley et al. [14] used CNN-extracted features instead of hand-crafted features due to the popularity and robustness of CNN. To learn document structures Roy et al. [15] proposed an ensemble model. In their model, DCNN models trained overall or specific regions of the documents, and the output of the models were ensembled with Support Vector Machine (SVM). Csurka [16] extracted shallow features from document images with RunLeght Histogram (RL) and Fisher Vector (FV) and then, used classical machine learning algorithms to classify these features. CNNs trained on ImageNet were applied with pre-processing, data augmentation, transfer learning, and domain adaptation to classify document images [17]. Yaman et al. [18] used fine-tuned deep CNN architectures namely VGG-16, AlexNet, and GoogLeNet to classify document images by using visual features. VGG-16 yielded results as the most successful method. In Afzal et al.'s study [19], AlexNet, VGG-16, GoogLeNet, and ResNet-50 architectures were applied by using transfer learning to compare the performance of large and small amounts of datasets. It was concluded that improper pre-training and small size training data cause poor performance. Zavallishin et al. [20] utilized text and non-text regions extracted with a novel Maximally Stable Extremal Regions (MSER)-based approach. In this study Grayscale Runlength Histogram (GRLH), Spatial Local Binary Pattern (SLBP), and Fisher Vectors based on Bernoulli Mixture Model (BMMFV) were used as visual features. On different datasets, SVM and majority voting algorithms were applied. Unlike other studies, document image classification was carried out in real-time in Kölsch et al.'s study [21]. The authors presented a two-stage classification mechanism. In the first stage, they utilized AlexNet which is good at extracting features and representing visually rich images. In the second stage, Extreme Learning Machines (ELMs) were fed with extracted features from the deep architecture. Das et al. [22] trained a stacked generalization-based ensemble of region-based VGG16 classifiers for classifying document images. The ensemble task was carried out over intra-domain transfer learning based sub-region similarity. Hassanpour and Malek [23] used ImageNet pretrained SqueezeNet to realize document image classification tasks by using features such as the title of documents, text alignment, tabular structures, and the presence of handwritten text. Mohsenzadegan et al. [24] applied pre-filtering methods to document images to create six different channels, then these channels were fed into CNN. With the usage of filtering methods, they obtained different features that made important contributions to classification. Siddiqui et al. [25] used self-supervised representation instead of the fine-tuning pre-trained model. The authors showed that, in the presence of limited labeled data, representations learnt using self-supervised representation learning approaches were an effective choice for document image classification. Shallow, deep, and very deep CNN models were tested on two classification tasks by Liu et al. [26], and it was found that shallow CNNs combined with pre-processing performed the best.

Şahin et al. [27] utilized only text to predict the classes of the document images. For this purpose, at first keywords for each class were extracted then words in OCRed Turkish documents were matched with class keywords. At the end, classes were determined according to the number of matches. Noce et al. [28] combined text and image-based approaches to improve classification accuracy. In their study, the structural embedding of textual topics obtained with OCR were represented with colored boxes and fed into the CNN for classification tasks. In Audebert et al.'s study [29], document images were classified by fusing textual and visual features based on an end-to-end learnable multimodal deep network model. While textual features were represented with FastText generated document embeddings, visual features were learned by using MobileNetv2, a state-of-the-art CNN architecture. Jain and Wigington [30] proposed two feature fusion mechanisms namely spatial fusion and feature fusion to classify document images. While in spatial fusion, image and spatial text embedding with VGG-16 were applied, in feature fusion, the fusion was realized from second layer to last layer in which image features were obtained from VGG-16 and text features were obtained from text ensemble network. When these two methods were compared, it was seen that the feature fusion method was more successful. Asim et al. [31] applied a simple ensemble of text and image models. In their model, from OCRed documents most important tokens were obtained with ranking and then these features were integrated into the CNN. The output of the CNN was combined with the output of the InceptionV3 which took images as input. Bakkali et al. [32] devised a deep two-headed architecture which was fed with OCR based textual and structural visual information obtained from document images simultaneously. In their cross-model, while semantic features were learned with BERT, visual features were learned with NASNet-Large pre-trained on ImageNet. Ferrando et al. [33] exploited from an ensemble of a BERT classifier and EfficientNets. LayoutLM was a multi-model pre-trained model which fused text and layout together for visually-rich documents. In LayoutLM [6], at first original BERT architecture was fed with tokens and their positions from scanned documents, then 2-D layout and image embeddings were integrated into it. Then the summed vector representations were integrated into multi-layer bidirectional transformer encoder to make classification. Cosma et al. [34] extracted topics from text with Latent Dirichlet Allocation (LDA) and then applied CNN to document images to predict these topics. In Bakkalli et al.'s study, [35] a hybrid-cross-modal end-to-end architecture was exploited to learn word-embeddings and structural information from document images simultaneously. These two obtained modalities were fused by using a late fusion scheme. Goodrum et al. [36] classified electronic health records by using OCR and text classification models. In another study, the authors presented an ensemble network that used a self-attention-based mutual learning with the Kullback–Leibler divergence regularization loss to jointly learn the visual structural and text embeddings from document images [37].

In this study, the fusion of content-based and image-based classifiers is carried out over a simple ensemble model. The ensemble process distinguishes this study from others. As an ensemble process, we utilize a weighted mean. In the weighted mean method, entropy is calculated using the probability distribution determined by the model for the classification labels and integrated into the method as a weight value. Even though we use ensemble learning as well, our research focuses on state-of-the-art machine learning and deep learning algorithms to demonstrate the utility of ensemble learning, particularly in document images classification.

3 Methodology

3.1 Content-based classification

3.1.1 Content-based features

In the traditional document classification task, the text of a document is characterized as a sequence of words, where each word is accepted as a discrete, unique element [38]. Since this study aims to classify document images where texts in such documents cannot be obtained directly, it is necessary to obtain texts by applying OCR. For this purpose, we used the LSTM based Tesseract OCR engine [39].

Powerful classification models need a strong and rich feature representation of data. In text classification tasks, the words in the documents are used as features of the text. After obtaining text with OCR, as the first step of feature extraction, pre-processing methods are applied. Special characters, punctuation, digits, and out-of-vocabulary (OOV) words are ignored. All characters are converted to lowercase because information classification methods are case-sensitive.

After applying the pre-processing steps, numerical representations of the texts are created by using the most popular document representation model Term Frequency-Inverse Document Frequency (TF-IDF) method. TF-IDF consists of two basic parts. Term frequency (TF) is the first of these parts. It expresses the frequency of the words in the document. The TF value of word w_i in $W = \{w_1, w_2, w_3, \ldots\}$ text is found by dividing the number of occurrences in the text by the total number of words. Inverse Document Frequency (IDF) is a value that is used to lower the TF frequent words in the dataset. Frequent words in the data set are less distinctive. Therefore, decreasing the TF values of these words affect the classification process positively. Document frequency (DF) is found by dividing the number of texts in which the word is found by the number of all texts for w_i. IDF is obtained by taking the logarithm of the DF value.

Because not all words are equally important, it's difficult to determine a document's class and significance simply on its content. Frequent words in texts with the same class label have a stronger representation ability for this study. Therefore, we extract word-frequency list $\{(w_1, f_1), (w_2, f_2), ...\}$ from texts with the same label information. However, because not all words are likely to be included in the feature set, we use a threshold-based pruning method. Every word w_i with a frequency f_i less than 10 is removed from the list. A part of the list is given in Table 1.

3.1.2 Traditional classification models

Traditional models are statistical in nature, relying on mathematical approaches to find specific patterns in large datasets. In this work, we used Multinomial Naive Bayes (MNB), SVM, Random Forest, Extra Tree, XGBoost, and Artificial Neural Network (ANN) models to realize content-based classification.

The Naive Bayes model, which is based on the Bayes theorem, is one of the first statistical models and has proven to be particularly successful and efficient for classification problems with large feature sets [40]. According to the approach, all the features $W = \{w_1, w_2, ..., w_n\}$ are conditionally independent given the class label C_i that is $P(W|C) = \prod_{i=1}^{n} P(w_i|C)$. MNB makes inferences from multinomial data for text classification. MNB is employed because of its simplicity of implementation and speed. In conditional probability, this strategy is based on the multinomial distribution [41].

The SVM separates data objects in a feature space into two classes across a decision boundary called a hyper-plane. While doing so, SVM aims to provide maximum separation between these two classes. The margin of the hyper-plane is the distance between two parallel hyper-planes that are equidistant from both sides of the hyper-plane and have no data items between them. When data items are not separated linearly, kernel functions are used to map the data items to a high-dimensional feature space [42]. The SVM model is developed for two-class problems. If extended to multi-class scenarios, one-versus-one (OVO) and one-versus-all (OVA) approaches have been proposed. OVO handles multi-class classification problems as a binary classification problem by dividing the dataset with each class against all other classes. Like OVO, OVA treats the multi-class classification problem as a binary classification problem, the difference is that OVA divides the dataset into one binary dataset for each class.

The Random Forest proposed by Breiman is a bagging ensemble model consisting of more than one decision tree [43]. The average of the estimation of the decision trees is taken as the estimation of the Random Forest model. The "Bootstrap" method is used for the training of decision trees. According to this method, each decision tree is trained with a subset of the dataset and repeat selections are possible. Also, a subset of the feature set is used when calculating the information gain for branching. All elements of the subset are chosen randomly.

Table 1 A part of the word-frequency list

Class label	Word-frequency list
ALES	('ales', 590), ('test', 458), ('number', 381), ('sayı',342), ('sonuç', 323), ...
Transcript	('ders', 445), ('akts', 434), ('aa', 347), ('kredi', 314), ('adı', 271), ...
Equivalence	('adı', 147), ('mezun', 144), ('diploma', 101), ('tarihi', 99), ('kimlik', 97), ...
...	...

Extra Tree is a classification model like Random Forest [44]. The most important difference from Random Forest is the usage of the entire dataset for the training of decision trees. Another difference between them is that the pivot point for branching is selected randomly.

The XGBoost model is an algorithm based on "gradient boosting" and devised for tree boosting [45]. XGBoost is a stronger learner by combining weak basic learning models in an iterative manner [46]. At each iteration of the gradient boost, the rest is used to correct the previous estimator, thus the loss function can be optimized. The most important characteristic of this decision tree-based method is that it is designed to run in a distributed environment. XGBoost gives very good results with large datasets.

ANNs are powerful learning models that mimic the behavior of living neuron cells. ANN models consist of connected sequential layers. A node in a layer is linked to all nodes in the previous layer. Weight values are assigned to links between nodes. While the input is transmitted from one layer to the next, values of input are multiplied by the weight values and these values are summed. Then, this sum is passed through the activation function and the output value is obtained. ANN models gain the ability to represent nonlinear patterns with activation functions.

3.2 CNN for image-based classification

When a visually rich document is given, it is inevitable to use visual information for classification. In this study, we use CNN models to make image-based classification and these selected models are applied under two different ways. First, hyper-parameters of models are adjusted based on experiments. Second, the models are created with the transfer learning approach. Pre-trained large models are very powerful and widely used for image classification problems. We select state of art models for handling problems.

3.2.1 Pre-processing

The resolution of the document images is 1170 x 827 which is too large to be given as input to the CNN. Such a large resolution causes both high computational costs and overfitting. Considering these problems, all images are resized to 500x300, 331x331 or 224x224 pixels by converting into 100 dots per inch (DPI) resolution. The details in the documents are ignored as a result, but the basic layout information is retained. The location of the tables, figures, texts, and other items is therefore required.

3.2.2 Convolutional neural network architectures

CNNs are very effective models for image classification problems. Much of the reason behind this CNNs do not require a feature engineering step. CNN models are generally composed of stacked convolutional layers that may be combined with a pooling layer and followed by several fully connected layers. Convolutional layers are the base components of CNN models, and these layers have learnable filters. Robust features are obtained from input with the help of filters. Pooling layers are placed between convolution layers to reduce the spatial size of obtained features, thus the number of parameters are reduced. Fully connected layers act as classifiers to classify features extracted by convolution layers.

The first model is constructed as a basic sequential convolution network. The network begins with two sequential convolution layers with twelve kernels of sizes 9 x 9 in order to extract deep features. The last stage is composed of 5 fully connected layers with nodes of 1024, 1024, 768, 512 and 12 respectively. While Rectified linear unit (ReLU) is used as an activation function in all the convolution layers, softmax is used as the activation function in the output of the model. To prevent loss of information, attention is paid to the width-height ratio. The image size is set to 500x300.

As a second model, we use Dense Convolutional Network (DenseNet) which is a deep CNN model with dense blocks composed of sequential convolutional layers [47]. The output of a convolution layer is concatenated with every other layer in a feed-forward fashion. To reduce the channel size of output of dense blocks and to make the model more compact, point-level convolution process is applied with a 1x1 filter. Thus, the number of parameters to be used is also reduced. With the dense blocks, the training success of the model increases. The proposed

DenseNet model which is called Concat-CNN consists of three dense blocks with three sequential convolution layers. Convolution layers contain 8 filters with the size of 3x3. The main difference from the original model is that the concatenation process does not connect the output of each layer to all other layers, but only to the fully connected layer. Namely, the output of the dense block $O_d \in R^{h,w,c_1+c_2+c_3}$ is obtained by concatenating the outputs of three layers $O_1 \in R^{h,w,c_1}$, $O_2 \in R^{h,w,c_2}$ and $O_3 \in R^{h,w,c_3}$. To reduce the size of block output, a convolution operation is applied with thirty-two 3x3 filters. ReLU activation function is used in all convolutional layers. Also, there are four fully connected layers which contain 512, 512, 256 and, 12 nodes respectively. The input size is set to 500x300 like the traditional CNN model. With the proposed architecture, the aim is to keep the performance of the model high and to try to minimize the physical resource consumption. The architecture of the Concat-CNN is given in Fig. 1.

3.2.3 Transfer learning

As the number of layers increases in convolutional networks, the representation capacity of the patterns increases. However, large models also come with the requirement of significant computation resources. In addition, these models need to be trained with large datasets. Therefore, many works in literature use pre-trained models. In particular, the usage of transfer learning from ImageNet for image classification problems is very common. In this work, we use NasNet-Large, InceptionV3, EfficientNet, and RasNet models with a transfer learning approach. A global pooling layer and a fully connected layer containing 12 nodes are added to make the models suitable for our classification problem.

The need for an optimized network has led developers to search for new ones. As a result of such a need, Neural Architecture Search Network (NASNet) emerged as a hybridization of AutoML and Neural Architectural Search (NAS). This optimized architecture was introduced by the Google Brain team in 2018 based on reinforcement learning. The purpose of NASNet is to reduce the number of parameters while increasing the classification performance. To achieve this, at first, they proposed a building block to search for the best cell on CIFAR-10 dataset then applied this cell to ImageNet [48]. In the NASNet architecture, only the best cells are selected instead of searching all cells. This will provide a faster search so that more generalized features can be obtained [49]. In this study, we propose the use of the NASNet which includes two specialized cells namely, Normal Cell and Reduction Cell. Normal cell equals the height and width dimension. The reduction cell reduces the height and width by two. So, the input size of 331 × 331 × 3 is reduced to a size of 42 × 42. Finally, in the output layer containing 12 nodes, the results of global pooling are combined.

InceptionV3 is a CNN model in the family of Inception models and the third generation of the GoogLeNet [50, 51]. It consists of inception modules that provide an increase in the efficiency of using model parameters and reduce computational cost with factorized convolutions and label smoothing. While factorized convolutional split large filter sizes into smaller convolutions, label smoothing achieves aggressive editing. In the experiments, 331x331x3 sized images are fed into InceptionV3 as input.

EfficientNet is a CNN model that was developed for mobile devices [52]. Model height and width and resolution of the image are used as parameters to develop an optimum model. EfficientNet has special blocks that contain deep-wise convolution and pointwise convolution layers. Therefore, these blocks have shortcut connections like ResNet models. There are 8 different EfficientNet models. All models are trained with the ImageNet dataset. These are powerful models with a very low number of parameters compared to other deep models. 331x331x3 sized images are used as input for this model.

Composed of residual blocks, Residual Networks (ResNet) is capable of training extremely deep networks [53]. In residual blocks, the output of convolutional layers is summed linearly. In other words, this arrangement resembles a shortcut connection. Thus, information flows between layers without the attenuation caused by a lot of computational burdens, resulting in enhanced optimization. ResNet takes 224x224x3 sized images for input.

3.3 Ensemble classification

Although both content-based and image-based classification achieve state-of-the-art results, making classification with only one kind of input is a barrier to the utilization of critical data. As a result, we recommend combining both content-based and image-based characteristics to achieve better outcomes. For this, we devise a novel weighted mean method that combines the predictions of content-based and image-based classification approaches. All classification models return the probability distribution of $C = c_1, c_2, \ldots, c_n$ class labels as a prediction. Our goal in performing the combination is to create a more stable model that will be more efficient in classification. Because if the model is unstable, prediction values in the distribution will be close to each other. As a result, in our proposed weighted mean method, entropy is used as a weight value. The entropy value increases as the distribution of prediction approaches the uniform distribution. That's why we

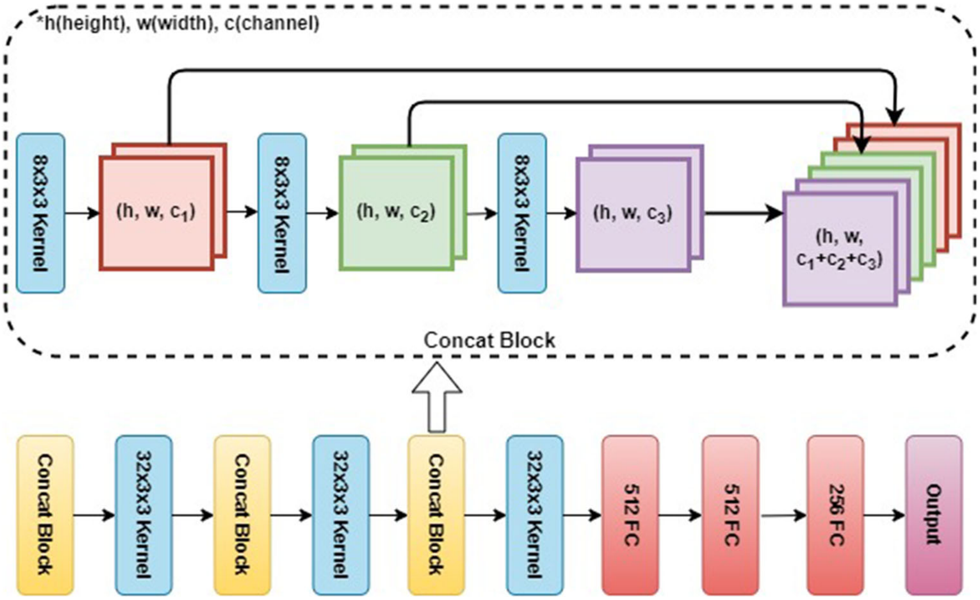


Fig. 1 Architecture of Concat-CNN model

compute the multiplicative inverse of entropy value. The complete pipeline is illustrated in Fig. 2.

$$w_i = \frac{1}{-\sum_j^n c_j log(c_j)}, j = 1, 2, ..., n \quad (1)$$

The weighted average of the probability distributions of C_1 and C_2 is calculated according to the formula below.

$$C_{mean} = \frac{w_1 C_1 + w_2 C_2}{w_1 + w_2} \quad (2)$$

4 Experiments and evaluation

4.1 Dataset

In the experiments, we use documents uploaded to the Kocaeli University's application system to apply for master's degree or undergraduate transfer between programs. The dataset consists of 1044 document images which are categorized into 12 classes. An example from each class in the dataset is given in the Fig. 3 and Table 2 gives a summary of all class labels. Due to the privacy of personal data, the picture, name, ID number etc. of the persons are marked in red.

4.2 Performance metrics

We calculate accuracy, precision, recall, and F-Score as performance metrics for document image classification. Where true positive (TP) is the correct classification of a document image otherwise, it is considered as false positive (FP). False negative (FN) is the incorrect classification of a document image as another type of document image. True negative (TN) is the correct classification of a document image as another type of document image.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \quad (3)$$

$$Precision = \frac{TP}{TP + FP} \quad (4)$$

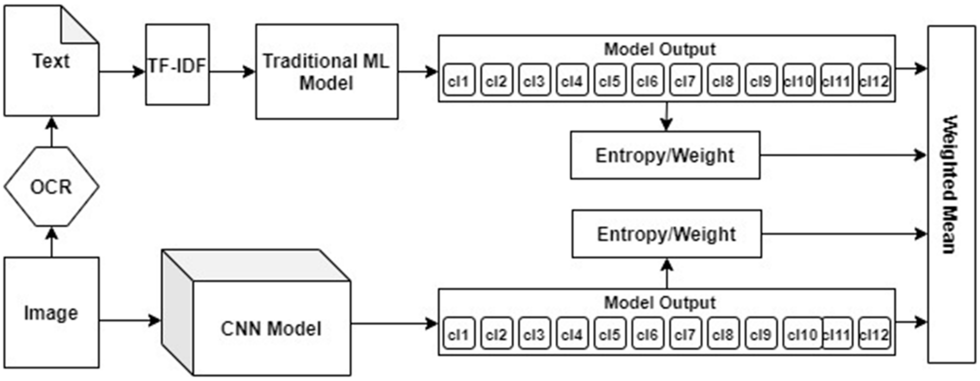


Fig. 2 Model architecture

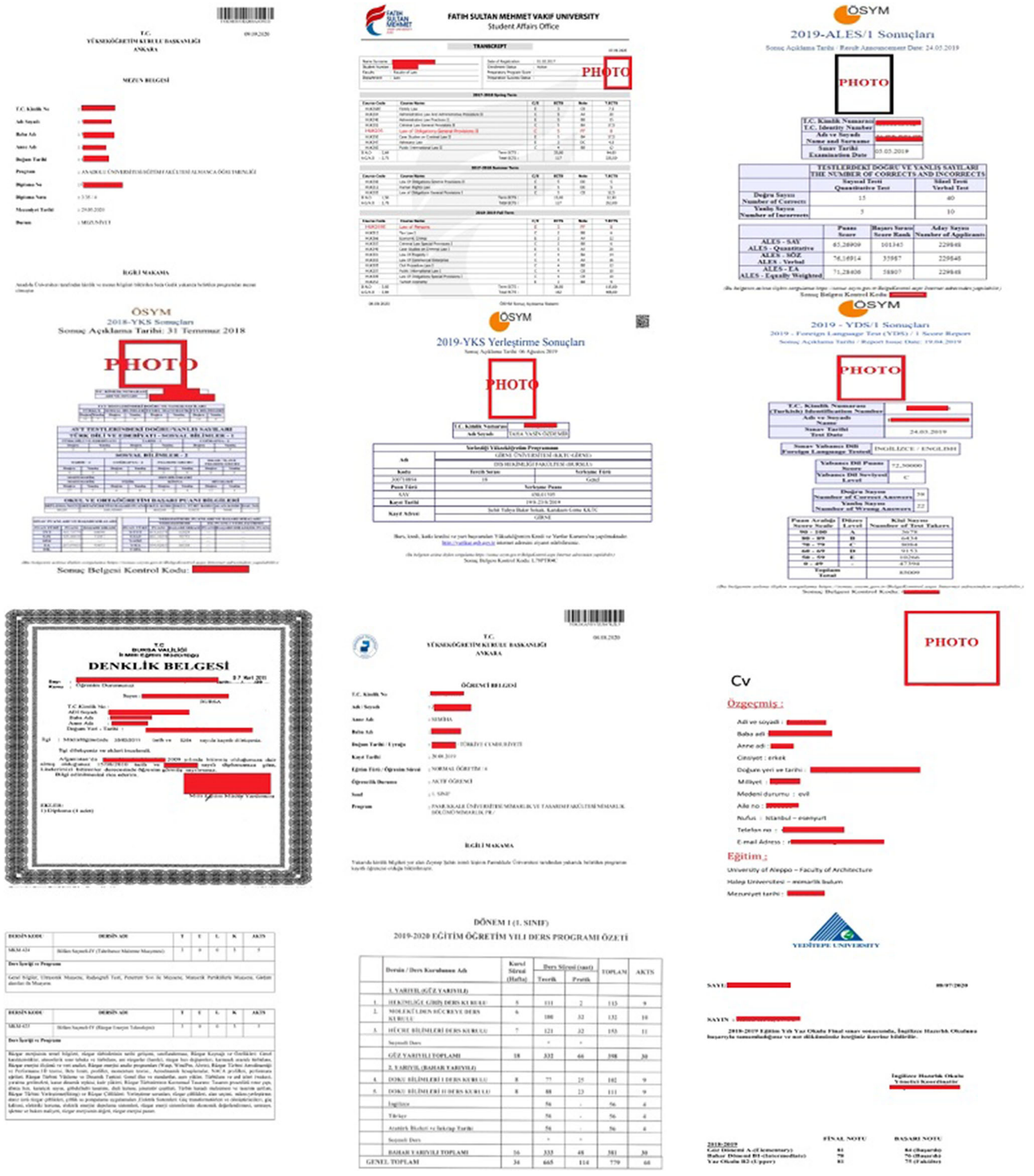

FATIH SULTAN MEHMET VAKIF UNIVERSITY
Student Affairs Office
TRANSCRIPT
PHOTO
ÖSYM
2019-ALES/1 Sonuçları
ÖSYM
2018-YKS Sonuçları
Sonuç Açıklama Tarihi: 31 Temmuz 2018
PHOTO
ÖSYM
2019-YKS Yerleştirme Sonuçları
PHOTO
ÖSYM
2019 - YDS/1 Sonuçları
PHOTO
DENKLİK BELGESİ
PHOTO
Cv
Özgeçmiş :
Eğitim :
YEDİTEPE UNIVERSITY
DÖNEM 1 (1. SINIF)
2019-2020 EĞİTİM ÖĞRETİM YILI DERS PROGRAMI ÖZETİ

Fig. 3 Document image samples from dataset

$$Recall = \frac{TP}{TP + FN} \tag{5}$$

$$F - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{6}$$

4.3 Experiments

Google Colaboratory (a.k.a Colab) is a popular development environment because of the hardware support it provides. Its goal is to encourage machine learning studies by supporting

Table 2 Summary of class labels

Document class	Number of documents
Diploma	100
Transcript	99
Academic personnel and postgraduate education entrance exam (ALES)	100
OSYM result of examination	49
OSYM placement document	58
Result of foreign language examination	98
Equivalence	71
Student certificate	100
CV	100
Course content	100
Course list	100
Preparatory certificate	69

Table 3 Summary of parameters of deep learning models

Model	Number of parameters	Fully connected layers	Optimizer	Learning rate	Number of epochs	Batch size
Traditional CNN	103 M	1024-1024-768-512-12	Adam	10^{-3}	40	32
Concat-CNN	722 K	512-512-256-12	Adam	25×10^{-5}	20	32
EfficientNetB3	10 M	15036-12	SGD	10^{-2}	10	16
InceptionV3	22 M	2048-12	SGD	10^{-2}	20	16
NasNet Large	85 M	4032-12	SGD	10^{-2}	17	16
ResNet50	24 M	2048-12	SGD	10^{-2}	17	16

Table 4 Results for traditional models on the test set- accuracy and F-score

Model	Accuracy	F-score
MNB	88.70%	88.73%
SVM	**91.26%**	**91.04%**
Random forest	90.37%	90.31%
Extra tree	91.15%	91.04%
XGBoost	90.37%	90.31%
ANN	90.96%	90.93%

The bold entries are necessary to show the best results

Table 5 Results for CNN models on the test set- accuracy and F-score

Model	Accuracy	F-score
Traditional CNN	76.42%	76.06%
Concat-CNN	**82.02%**	**82.23%**

The bold entries are necessary to show the best results

Table 6 Results for transfer learning models on the test set- accuracy and F-score

Model	Accuracy	F-score
EfficientNetB0	90.27%	90.35%
EfficientNetB1	90.76%	90.76%
EfficientNetB2	91.45%	91.46%
EfficientNetB3	**92.04%**	**92.09%**
EfficientNetB4	91.65%	91.67%
EfficientNetB5	87.13%	87.47%
EfficientNetB6	84.31%	84.64%
EfficientNetB7	91.55%	91.44%
InceptionV3	91.45%	91.41%
NasNetLarge	91.45%	91.47%
ResNet50	91.06%	91.04%

The bold entries are necessary to show the best results

Table 7 Results for ensemble models on the test set - accuracy and F-score

Model1	Model2	Accuracy	F-score
Traditional CNN	MNB	87.62%	87.28%
	SVM	90.57%	90.39%
	Random forest	87.92%	87.53%
	Extra tree	88.11%	87.82%
	XGBoost	87.92%	87.53%
	ANN	84.48%	84.42%
Concat-CNN	MNB	89.00%	89.07%
	SVM	90.47%	90.55%
	Random forest	88.80%	88.81%
	Extra tree	89.78%	89.87%
	XGBoost	88.80%	88.81%
	ANN	87.03%	87.13%
EfficientNetB3	MNB	93.52%	93.60%
	SVM	94.30%	94.38%
	Random forest	94.20%	94.23%
	Extra tree	**94.40%**	**94.45%**
	XGBoost	94.20%	94.23%
	ANN	93.22%	93.24%
InceptionV3	MNB	92.43%	92.38%
	SVM	93.81%	93.74%
	Random forest	93.12%	93.02%
	Extra tree	93.61%	93.54%
	XGBoost	93.12%	93.02%
	ANN	91.84%	91.92%
NasNET large	MNB	92.43%	92.47%
	SVM	93.32%	93.39%
	Random forest	92.92%	92.95%
	Extra tree	93.12%	93.19%
	XGBoost	92.92%	92.95%
	ANN	91.74%	91.86%
ResNet50	MNB	91.35%	91.08%
	SVM	93.02%	93.08%
	Random forest	92.04%	91.93%
	Extra tree	92.53%	92.47%
	XGBoost	92.04%	91.93%
	ANN	91.35%	91.08%

The bold entries are necessary to show the best results many libraries [54]. Tensorflow and Scikit-learn are two of the libraries available for creating classification models [55, 56].

Tensorflow is often preferred in deep learning applications. Likewise, Scikit-Learn is a widely used library for classification, clustering and regression problems. In the experiments, Scikit-Learn is used for content-based models, Tensorflow is used for image-based models. As an exceptional case, Tensorflow is also used for the neural network model.

Since the class distributions in our document image dataset are imbalanced, we apply 10-fold cross-validation to avoid bias in the performances of the models due to skewness. Each iteration, a separate one-fold is utilized as a test set, for a total of 10 iterations, with the remaining 9-folds serving as the training set.

Since there are 12 different class labels in our dataset, we must use the multi-label smoothing regularization to improve classification performance by allocating a uniform label distribution to multi-label training set. For this purpose, various experiments are carried out for the data set and the smoothing parameter is set to 0.2. Since the dataset has 12 different classes, the SVM model is created with the OVA approach. Linear kernel is used to linearly separate the samples in the dataset. C parameter, which determines the width of the hyperplane in the model and is used to avoid overfitting and underfitting situations, is set to 100. The Random Forest model is created using 200 decision trees. The size of the feature subset selection for branching is assigned as the square root of the number of all features. Gini Index is used to calculate the information gain. The Extra Tree model is created with 50 decision tree classifiers. The size of feature subsets randomly selected within the feature set is determined as 10% of the number of all remaining features. Gini Index is used to calculate the information gain. The XGBoost model is created with 50 basic classifiers and the depth of the model is limited to 6. The learning rate is set to 0.3 for parameter optimization. Ridge Regularization is used in the model to prevent overfitting and the 2.11 is assigned to the lambda parameter. Adam optimization algorithm is used to optimize parameters of ANN. Adam gives good results for sparse-matrix inputs like the TF-IDF matrix. Categorical cross-entropy is used as the loss function. The learning rate is set to 10^{-3}. The number of trainable parameters of the model is 546,816. Model training lasts 30 epochs with 32 batch sizes. The model consists of an input layer containing 544 nodes, two hidden layers containing 512 nodes, and an output layer containing 12 nodes. The summary of parameters of deep learning models are given in Table 3. Table 4 shows the results of the models on the test set in terms of accuracy and F-score. The models indicated as bold in the tables are superior models. When the performances of traditional models are compared, it is seen that the best result is produced by SVM because of its generalization ability. The worst result is obtained in the MNB among the models, while the results of the other models are close to each other.

To train the Traditional CNN model and Concat-CNN model, the Adam algorithm is used. Categorical cross-entropy is chosen as the loss function for both models. The categorical cross-entropy calculates the loss of an data sample as follow:

$$Loss = -\sum_{i=1}^{n} y_i log \hat{y}_i \tag{7}$$

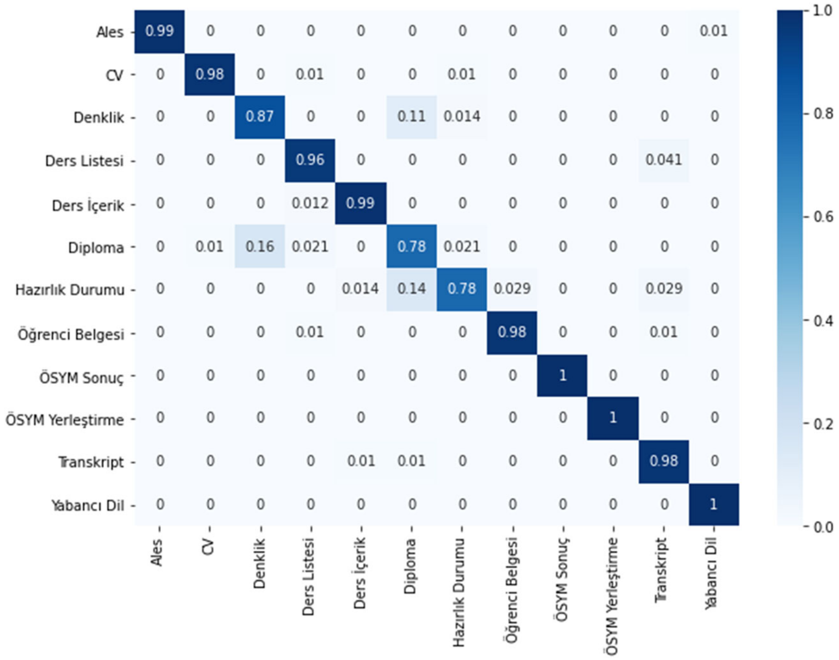
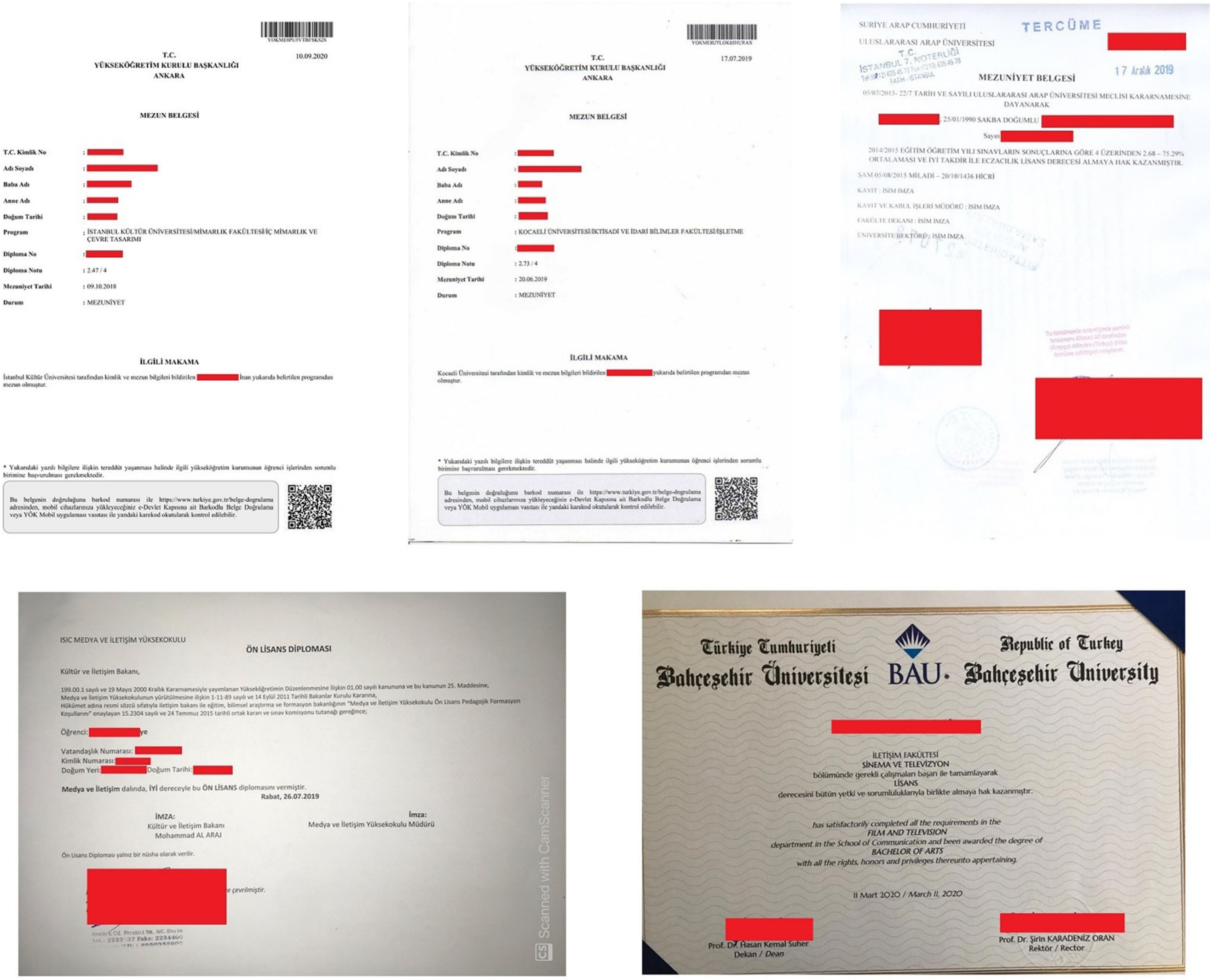


Fig. 4 Class-confusion matrix for document image classification

where y_i is the expected value, $\hat{y}_i$ is observed value of the ith data sample, and n is the number of class.

While the learning rate for the traditional CNN model is 10^{-3}, the learning rate for the Concat-CNN model is assigned as 25×10^{-5}. Traditional CNN model has 102,934,390 parameters. The number of epochs is assigned as 40 for the traditional model. The Concat-CNN model has 721,532 parameters and is trained with 20 epochs. The batch size is chosen as 32 for both models. The number of trainable parameters of the traditional CNN is very large compared to the Concat-CNN. However, better results are obtained from the Concat-CNN than the traditional CNN model. The performances of both models remain weak compared to the content-based models. Table 5 shows the results of the models on the test set in terms of accuracy and F-score. Bold ones in table represent the best score.

SGD algorithm is used to train all transfer learning models with a 0.01 learning rate. Categorical cross-entropy is chosen as the loss function for each model. The batch size is set to 16. The number of total trainable parameters is 84,768,546 in the NasNet Large model. The maximum test performance is achieved by training with 17 epochs. The InceptionV3 model has 21,792,940 trainable parameters and is trained with 20 epochs. EfficientNet is trained with 10 epochs, and it has a total 10,714,676 trainable parameters. The EfficientNetB3 model is superior to the other EfficientNet models due to both accuracy and F-score. ResNet50 has 23,559,180 trainable parameters and is trained with



YOKMESPU5VTBFSKS2S

10.09.2020

T.C.
YÜKSEKÖĞRETİM KURULU BAŞKANLIĞI
ANKARA

MEZUN BELGESİ

T.C. Kimlik No : [redacted]
Adı Soyadı : [redacted]
Baba Adı : [redacted]
Anne Adı : [redacted]
Doğum Tarihi : [redacted]
Program : İSTANBUL KÜLTÜR ÜNİVERSİTESİ/MİMARLIK FAKÜLTESİ/İÇ MİMARLIK VE ÇEVRE TASARIMI
Diploma No : [redacted]
Diploma Notu : 2.47 / 4
Mezuniyet Tarihi : 09.10.2018
Durum : MEZUNİYET

İLGİLİ MAKAMA

İstanbul Kültür Üniversitesi tarafından kimlik ve mezun bilgileri bildirilen [redacted]İnan yukarıda belirtilen programdan mezun olmuştur.

* Yukarıdaki yazılı bilgilere ilişkin tereddüt yaşanması halinde ilgili yükseköğretim kurumunun öğrenci işlerinden sorumlu birimine başvurulması gerekmektedir.

Bu belgenin doğruluğunu barkod numarası ile https://www.turkiye.gov.tr/belge-dogrulama adresinden, mobil cihazlarınıza yükleyeceğiniz e-Devlet Kapısına ait Barkodlu Belge Doğrulama veya YÖK Mobil uygulaması vasıtası ile yandaki karekod okutularak kontrol edilebilir.

YOKMEBUTLOK8SHURAX

17.07.2019

T.C.
YÜKSEKÖĞRETİM KURULU BAŞKANLIĞI
ANKARA

MEZUN BELGESİ

T.C. Kimlik No : [redacted]
Adı Soyadı : [redacted]
Baba Adı : [redacted]
Anne Adı : [redacted]
Doğum Tarihi : [redacted]
Program : KOCAELİ ÜNİVERSİTESİ/İKTİSADİ VE İDARİ BİLİMLER FAKÜLTESİ/İŞLETME
Diploma No : [redacted]
Diploma Notu : 2.73 / 4
Mezuniyet Tarihi : 20.06.2019
Durum : MEZUNİYET

İLGİLİ MAKAMA

Kocaeli Üniversitesi tarafından kimlik ve mezun bilgileri bildirilen [redacted]yukarıda belirtilen programdan mezun olmuştur.

* Yukarıdaki yazılı bilgilere ilişkin tereddüt yaşanması halinde ilgili yükseköğretim kurumunun öğrenci işlerinden sorumlu birimine başvurulması gerekmektedir.

Bu belgenin doğruluğunu barkod numarası ile https://www.turkiye.gov.tr/belge-dogrulama adresinden, mobil cihazlarınıza yükleyeceğiniz e-Devlet Kapısına ait Barkodlu Belge Doğrulama veya YÖK Mobil uygulaması vasıtası ile yandaki karekod okutularak kontrol edilebilir.

SURİYE ARAP CUMHURİYETİ
ULUSLARARASI ARAP ÜNİVERSİTESİ

TERCÜME

[redacted]

T.C.
İSTANBUL 7. NOTERLİĞİ
Tel:(0212) 635 45 72 Fax:(0212) 635 45 76
FATİH - İSTANBUL

17 Aralık 2019

MEZUNİYET BELGESİ

05/07/2015- 22/7 TARİH VE SAYILI ULUSLARARASI ARAP ÜNİVERSİTESİ MECLİSİ KARARNAMESİNE DAYANARAK

[redacted], 25/01/1990 SAKBA DOĞUMLU [redacted]

Sayın [redacted]

2014/2015 EĞİTİM ÖĞRETİM YILI SINAVLARIN SONUÇLARINA GÖRE 4 ÜZERİNDEN 2.68 – 75.29% ORTALAMASI VE İYİ TAKDİR İLE ECZACILIK LİSANS DERECESİ ALMAYA HAK KAZANMIŞTIR.

ŞAM 05/08/2015 MİLADİ – 20/10/1436 HİCRİ

KAYIT : İSİM İMZA

KAYIT VE KABUL İŞLERİ MÜDÜRÜ : İSİM İMZA

FAKÜLTE DEKANI : İSİM İMZA

ÜNİVERSİTE REKTÖRÜ : İSİM İMZA

[redacted]

Bu tercümenin noterliğimiz yeminli tercümanı Ahmad Ali tarafından (Arapça) dilinden (Türkçe) diline tercüme edildiğini onaylarım.

[redacted]

ISIC MEDYA VE İLETİŞİM YÜKSEKOKULU

ÖN LİSANS DİPLOMASI

Kültür ve İletişim Bakanı,

199.00.1 sayılı ve 19 Mayıs 2000 Krallık Kararnamesiyle yayımlanan Yükseköğretimin Düzenlenmesine ilişkin 01.00 sayılı kanununa ve bu kanunun 25. Maddesine,
Medya ve İletişim Yüksekokulunun yürütülmesine ilişkin 1-11-89 sayılı ve 14 Eylül 2011 Tarihli Bakanlar Kurulu Kararına,
Hükümet adına resmi sözcü sıfatıyla iletişim bakanı ile eğitim, bilimsel araştırma ve formasyon bakanlığının "Medya ve İletişim Yüksekokulu Ön Lisans Pedagojik Formasyon Koşullarını" onaylayan 15.2304 sayılı ve 24 Temmuz 2015 tarihli ortak kararı ve sınav komisyonu tutanağı gereğince;

Öğrenci: [redacted]ye

Vatandaşlık Numarası: [redacted]
Kimlik Numarası: [redacted]
Doğum Yeri: [redacted] Doğum Tarihi: [redacted]

Medya ve İletişim dalında, **İYİ** dereceyle bu **ÖN LİSANS** diplomasını vermiştir.
Rabat, 26.07.2019

İMZA:
Kültür ve İletişim Bakanı
Mohammad AL ARAJ

İmza:
Medya ve İletişim Yüksekokulu Müdürü

Ön Lisans Diploması yalnız bir nüsha olarak verilir.

[redacted] e çevrilmiştir.

Atatürk Cd. Feraizci Sk. 8/C-Bursa
Tel.: 2222 37 Faks: 2234409

Türkiye Cumhuriyeti
Bahçeşehir Üniversitesi

BAU.

Republic of Turkey
Bahçeşehir University

[redacted]

İLETİŞİM FAKÜLTESİ
SİNEMA VE TELEVİZYON
bölümünde gerekli çalışmaları başarı ile tamamlayarak
LİSANS
derecesini bütün yetki ve sorumluluklarıyla birlikte almaya hak kazanmıştır.

has satisfactorily completed all the requirements in the
FILM AND TELEVISION
department in the School of Communication and been awarded the degree of
BACHELOR OF ARTS
with all the rights, honors and privileges thereunto appertaining.

11 Mart 2020 / *March 11, 2020*

[redacted]
Prof. Dr. Hasan Kemal Suher
Dekan / *Dean*

[redacted]
Prof. Dr. Şirin KARADENİZ ORAN
Rektör / Rector

Fig. 5 Document images of the diploma class

20 epochs. When all the experiments are considered, the best performance is achieved with the EfficientNetB3 model. The reason for the success of the EfficientNet architecture is that it scales across all depth, width, and resolution dimensions using a compound coefficient [57].

The performances of transfer learning models remain weak compared to the content-based models. Table 6 shows the results of the models on the test set in terms of accuracy and F-score. Bold ones in table represent the best score.

To combine the results of content-based models and image-based models, the weighted average is used. Since the traditional CNN model has the lowest performance among other models, it is not included in the voting process. Also, traditional and Concat-CNN have been shown to reduce the success of content-based classifiers.

Among the combined test results, we obtained the best result from the EfficientNetB3 and Extra Tree duo. Table 7 shows the results of the ensemble models on the test set in terms of accuracy and F-score. Bold ones in table represent the best scores. When the results are checked in detail, it is realized that EfficientNetB3 is the one that improves the performance of content-based models the most among other methods. Class-confusion matrix for document image classification with EfficientNetB3 and Extra Tree duo on the dataset is given in Fig. 4.

The experimental results show that the performance of the ensemble of transfer learning models and traditional classification models is better and comparable with the state of the arts. The reason behind the higher performance is due to the feature representation strategy of the models and the combination of these strategies in the ensemble.

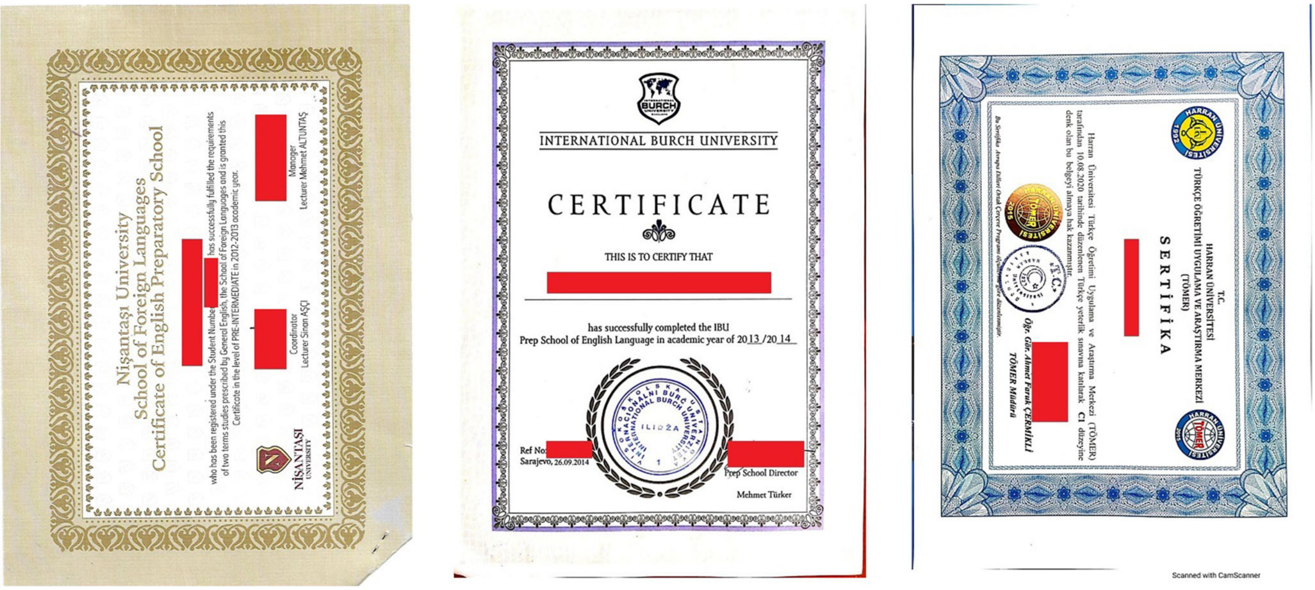
One of the reasons for the low performance of the Concat-CNN is that the model is trained from scratch. However, the results are remarkable considering the amount of data available. It is also the computationally light in terms of runtime overhead. When the results are combined with traditional methods using the ensemble method, performance is comparable to that of other models. Due to the low resource consumption, the usability of the model in real life problems is higher than other models.

Considering the test results, it has been observed that the EfficientNetB3, which is the most successful model in document image classification, cannot fully distinguish between certain classes.

The document image samples given in Fig. 5 are test images belong to the diploma class which have been missclassified as equivalence documents by the EfficientNetB3 model. When the documents belonging to the diploma and equivalence classes are examined, it is seen that they are structurally very close to each other. Therefore, it becomes difficult for the model to distinguish between classes. In addition, the variability in the structure of the samples in the diploma class makes it difficult for the model to learn visual patterns for prediction.

The other two classes in which the model has difficulty in making the distinction are diploma and preparatory certificate classes (Fig. 6). Preparatory certificates contain visual structures that are structurally similar to diploma documents. This situation makes it difficult to distinguish the examples belonging to the preparatory certificate class from the examples belonging to the diploma class.

In addition to the performance of the models, the execution times should also be taken into account. Therefore, the execution times of all models are obtained by setting the hardware accelerator parameter to none and GPU in Google Colaboratory environment and running times are shown in Table 8.



Nişantaşı University
School of Foreign Languages
Certificate of English Preparatory School

[redacted]

who has been registered under the Student Numbe[redacted] has successfully fulfilled the requirements of two terms studies prescribed by General English, the School of Foreign Languages and is granted this Certificate in the level of PRE-INTERMEDIATE in 2012-2013 academic year.

NİŞANTAŞI
UNIVERSITY

[redacted]
Coordinator
Lecturer Sinan AŞÇI

[redacted]
Manager
Lecturer Mehmet ALTUNTAŞ

BURCH
INTERNATIONAL BURCH UNIVERSITY

CERTIFICATE

THIS IS TO CERTIFY THAT

[redacted]

has successfully completed the IBU
Prep School of English Language in academic year of 2013 /2014

INTERNACIONALNI BURČ UNIVERZITET
INTERNATIONAL BURCH UNIVERSITY
ILIDŽA

Ref No:[redacted]
Sarajevo, 26.09.2014

[redacted]
Prep School Director
Mehmet Türker

T.C.
HARRAN ÜNİVERSİTESİ
TÜRKÇE ÖĞRETİMİ UYGULAMA VE ARAŞTIRMA MERKEZİ
(TÖMER)

SERTİFİKA

[redacted]

Harran Üniversitesi Türkçe Öğretimi Uygulama ve Araştırma Merkezi (TÖMER) tarafından 10.08.2020 tarihinde düzenlenen Türkçe yeterlik sınavına katılarak C1 düzeyine denk olan bu belgeyi almaya hak kazanmıştır.

[redacted]
Öğr. Gör. Ahmet Faruk ÇERMİKLİ
TÖMER Müdürü

Bu Sertifika Avrupa Dilleri Ortak Çerçeve Programı ölçütlerine göre düzenlenmiştir.

Fig. 6 Document images of the preparatory certificate class

Table 8 Comparison of running times of the models

Model	None (second)	GPU (second)
Traditional CNN	7.31	0.17
Concat-CNN	6.76	0.26
EfficientNetB3	26.44	0.85
InceptionV3	26.15	0.6
NasNetLarge	138.18	2.73
ResNet50	16.64	0.69

When the running times of the models are compared, it is seen that the NasNet Large model has the highest running time. It has been observed that while the number of parameters used by model decreases; the running time also decreases. However, the running times of the traditional CNN model with the highest number of parameters and the Concat-CNN model with the least number of parameters are close to each other. The main reason for this is the differences between the architectures. The traditional CNN model has a linear architecture. In addition, the number of convolution layers used in the model is less than in other models. Other models have more convolution layers and the connections between these layers are more complex. As a result of the study, it was concluded that the convolution process and the connection type between the layers are the main factors affecting the running time.

5 Conclusions

In this paper, we presented a digital document classification system that classifies the documents belonging to Kocaeli University Student Affairs Department. The textual features obtained from the content by OCR and the image-based features extracted with CNN-based models are used in the representation of the documents. Multiple machine learning and deep learning methods are explored to achieve accurate classification results. Apart from this, to offer a more effective solution; according to the obtained experimental performances of both document representation and classification results; we propose a document image classification method that fuses content-based and image-based models. The entropy values of returned probability distributions from classification models are used as weight values in the fusion method which is based on the weighted mean.

To sum up, SVM is the most successful model among the content-based models. EfficientNetB3 is the superior model over the CNN models and transfer learning models that learn image features of documents images. Given the ensemble models' results, these conclusions remain unaltered. The best classification performance is from the EfficientNetB3 and Extra Tree duo. As a future work we plan to carry out different fusion mechanisms of the presented models.

Acknowledgements We thank Ayhan Gültekin (Kocaeli University) for providing the experimental dataset.

Funding This work has been supported by the Kocaeli University Scientific Researchand Development Support Program (BAP) in Turkey under project number FBA-2020-2152.

Declarations

Conflict of Interests The authors declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

References

1. Mahajan K, Sharma M, Vig L (2019) Character keypoint-based homography estimation in scanned documents for efficient information extraction. In: 2019 international conference on document analysis and recognition workshops (ICDARW), vol 4. IEEE, pp 25–30
2. Menhour H et al (2021) Searchable Turkish OCRed historical newspaper collection 1928–1942. In: Journal of information science. SAGE Publications Sage UK, London, England, p 01655515211000642
3. Eken S, Menhour H, Küksal K (2019) DoCA: a content-based automatic classification system over digital documents. IEEE Access 7:97996–98004
4. Blanke T, Bryant M, Hedges M (2012) Ocropodium: open source OCR for small-scale historical archives. J Inf Sci 38(1):76–86
5. Hua Y et al (2020) Attention-based graph neural network with global context awareness for document understanding. In: China national conference on Chinese computational linguistics, Springer, pp 45–56
6. Xu Y et al (2020) Layoutlm: pre-training of text and layout for document image understanding. In: Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & Data Mining, pp 1192–1200
7. Mathew M et al (2021) Asking questions on handwritten document collections. Int J Doc Anal Recog (IJDAR) 24(3):235–249
8. Elanwar R et al (2021) Extracting text from scanned Arabic books: a large-scale benchmark dataset and a fine-tuned Faster-R-CNN model. Int J Doc Anal Recog (IJDAR) 24(4):349–362
9. Liu L et al (2021) Document image classification: progress over two decades. Neurocomputing 453:223–240
10. Rouhou AC et al (2022) Transformer-based approach for joint handwriting and named entity recognition in historical document. Pattern Recog Lett 155:128–134
11. Kumar J, Ye P, Doermann D (2014) Structural similarity for document image classification and retrieval. Pattern Recog Lett 43:119–126
12. Kang L et al (2014) Convolutional neural networks for document image classification. In: 2014 22nd international conference on pattern recognition, IEEE, pp –3172
13. Afzal MZ et al (2015) Deepdocclassifier: document classification with deep convolutional neural network. In: 2015 13th international conference on document analysis and recognition (ICDAR), IEEE, pp 1111–1115

14. Harley AW, Ufkes A, Derpanis KG (2015) Evaluation of deep convolutional nets for document image classification and retrieval. In: 2015 13th international conference on document analysis and recognition (ICDAR), IEEE, pp 991–995
15. Roy S, Das A, Bhattacharya U (2016) Generalized stacking of layerwise-trained deep convolutional neural networks for document image classification. In: 2016 23rd international conference on pattern recognition (ICPR), IEEE, pp 1273–1278
16. Csurka G (2017) Document image classification, with a specific view on applications of patent images. In: Current challenges in patent information retrieval, Springer, pp 325–350
17. Tensmeyer C, Martinez T (2017) Analysis of convolutional neural networks for document image classification. In: 2017 14th IAPR international conference on document analysis and recognition (ICDAR), vol 1. IEEE, pp 388–393
18. Yaman D, Eyiokur FI, Ekenel HK (2017) Comparison of convolutional neural network models for document image classification. In: 2017 25th signal processing and communications applications conference (SIU), IEEE, pp 1–4
19. Afzal MZ et al (2017) Cutting the error by half: investigation of very deep cnn and advanced training strategies for document image classification. In: 2017 14th IAPR international conference on document analysis and recognition (ICDAR), vol 1. IEEE, pp 883–888
20. Zavalishin S et al (2017) Document image classification on the basis of layout information. Electr Imaging 2017(2):78–86
21. Kölsch A et al (2017) Real-time document image classification using deep CNN and extreme learning machines. In: 2017 14th IAPR international conference on document analysis and recognition (ICDAR), vol 1. IEEE, pp 1318–1323
22. Das A et al (2018) Document image classification with intra-domain transfer learning and stacked generalization of deep convolutional neural networks. In: 2018 24th international conference on pattern recognition (ICPR), IEEE, pp 3180–3185
23. Hassanpour M, Malek H (2019) Document Image Classification using SqueezeNet Convolutional Neural Network. In: 2019 5th Iranian conference on signal processing and intelligent systems (ICSPIS), IEEE, pp 1–4
24. Mohsenzadegan K et al (2020) A convolutional neural network model for robust classification of document-images under real-world hard conditions. In: Developments of artificial intelligence technologies in computation and robotics: proceedings of the 14th international FLINS conference (FLINS 2020), World Scientific, pp 1023–1030
25. Siddiqui SA, Dengel A, Ahmed S (2021) Self-supervised representation learning for document image classification. IEEE Access 9:164358–164367
26. Liu Y, Soh L-K, Lorang E (2021) Investigating coupling preprocessing with shallow and deep convolutional neural networks in document image classification. J Electron Imaging 30(4):043024
27. Şahin S et al (2020) Dijital Dokümanların Anahtar Kelime Tabanlı Doğrulanması. In: Proceedings of the 6. Ulusal Yüksek Başarımlı Hesaplama Konferansı (in Turkish), pp 1–6
28. Noce L et al (2016) Embedded textual content for document image classification with convolutional neural networks. In: Proceedings of the 2016 ACM symposium on document engineering, pp 165–173
29. Audebert N et al (2019) Multimodal deep networks for text and image-based document classification. In: Joint european conference on machine learning and knowledge discovery in databases, Springer, pp 427–443
30. Jain R, Wigington C (2019) Multimodal Document Image Classification. In: 2019 international conference on document analysis and recognition (ICDAR), IEEE, pp 71–77
31. Asim MN et al (2019) Two stream deep network for document image classification. In: 2019 international conference on document analysis and recognition (ICDAR), IEEE, pp 1410–1416
32. Bakkali S et al (2020) Cross-modal deep networks for document image classification. In: 2020 ieee international conference on image processing (ICIP), IEEE, pp 2556–2560
33. Ferrando J et al (2020) Improving accuracy and speeding up Document Image Classification through parallel systems. In: International conference on computational science, Springer, pp 387–400
34. Cosma A et al (2020) Self-supervised Representation Learning on Document Images. In: International workshop on document analysis systems, Springer, pp 103–117
35. Bakkali S et al (2020) Visual and textual deep feature fusion for document image classification. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops, pp 562–563
36. Goodrum H, Roberts K, Bernstam EV (2020) Automatic classification of scanned electronic health record documents. Int J Med Inf 144:104302
37. Bakkali S et al (2021) EAML: ensemble self-attention-based mutual learning network for document image classification. Int J Doc Anal Recog (IJDAR) 24(3):251–268
38. Prieto JR et al (2021) Textual-content-based classification of bundles of untranscribed manuscript images
39. Kay A (2007) Tesseract: an open-source optical character recognition engine. Linux J 2007(159):2
40. Tang B, Kay S, He H (2016) Toward optimal feature selection in naive Bayes for text categorization. IEEE Trans Knowl Data Eng 28(9):2508–2521
41. Farisi AA, Sibaroni Y, Al Faraby S (2019) Sentiment analysis on hotel reviews using Multinomial Naıve Bayes classifier. J Phys Conf Ser 1192(1):012024
42. Goudjil M et al (2018) A novel active learning method using SVM for text classification. Int J Autom Comput 15(3):290–298
43. Breiman L (2001) Random forests. Mach Learn 45(1):5–32
44. Geurts P, Ernst D, Wehenkel L (2006) Extremely randomized trees. Mach Learn 63(1):3–42
45. Chen T, Guestrin C (2016) Xgboost: a scalable tree boosting system. In: Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining, pp 785–794
46. Friedman JH (2001) Greedy function approximation: a gradient boosting machine. Ann Stat 29(5):1189–1232
47. Huang G et al (2017) Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4700–4708
48. Zoph B et al (2018) Learning transferable architectures for scalable image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 8697–8710
49. Radhika K et al (2020) Performance analysis of NASNet on unconstrained ear recognition. In: Nature inspired computing for data science, Springer, pp 57–82
50. Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z (2016) Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2818–2826

51. Szegedy C et al (2015) Going deeper with convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1–9
52. Tan M, Le Q (2019) Efficientnet: rethinking model scaling for convolutional neural networks. In: International conference on machine learning, PMLR, pp 6105–6114
53. He K et al (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778
54. Prashanth B, Mendu M, Thallapalli R (2021) Cloud based Machine learning with advanced predictive Analytics using Google Colaboratory. Materials today: proceedings, Elsevier
55. Tensorflow. https://www.tensorflow.org/. Accessed 18 June 2021
56. ScikitLearn. https://scikit-learn.org/stable/. Accessed 18 June 2021
57. Zhang J, Guo H, Chen Z (2021) A concatenated approach based on transfer learning and PCA for classifying bees and wasps. J Phys Conf Ser 1873(1):012058

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Affiliations

Sevinç İlhan Omurca[1] · Ekin Ekinci[2] · Semih Sevim[3] · Eren Berk Edinç[1] · Süleyman Eken[4] · Ahmet Sayar[1]

Sevinç İlhan Omurca
silhan@kocaeli.edu.tr

Semih Sevim
ssevim@bandima.edu.tr

Eren Berk Edinç
erenberkedinc@hotmail.com

Süleyman Eken
suleyman.eken@kocaeli.edu.tr

Ahmet Sayar
ahmet.sayar@kocaeli.edu.tr

[1] Department of Computer Engineering, Kocaeli University, Kocaeli, Turkey

[2] Department of Computer Engineering, Sakarya University of Applied Sciences, Sakarya, Turkey

[3] Department of Computer Engineering, Bandırma Onyedi Eylül University, Balıkesir, Turkey

[4] Department of Information Systems Engineering, Kocaeli University, Kocaeli, Turkey