

# Study of Feature Fusion Methods for Document Image Classification

Jason Ravagli  
Università degli Studi di Firenze  
Florence, Italy  
jason.ravagli@stud.unifi.it

Iacopo Erpichini  
Università degli Studi di Firenze  
Florence, Italy  
iacopo.erpichini@stud.unifi.it

**Abstract**—State-of-the-art methods for document image classification rely on visual features extracted by deep convolutional neural networks (CNNs). These methods do not use rich semantic information present in the text of the document, which can be extracted using Optical Character Recognition (OCR). In this work we study a feature fusion technique original proposed by R. Jain and C. Wigington [1]. It allows to build a model that considers both visual and textual features of a document image to perform the classification. We compare this method to other approaches for the document image classification task using the RVL-CDIP dataset [2].

**Index Terms**—Document Image Classification, Text Classification, CNN, Transfer Learning, Feature Fusion

## I. INTRODUCTION

Automated document image classification can be a very important task in various contexts. For example can be crucial inside a document management system, to help search and information retrieval from a collection of documents. Furthermore, classify a document image could be the first important step of a Document Image Processing Pipeline, where knowing the type of document can guide the processes of information extraction.

For these reasons, many studies have been carried out on this topic. However, most of them focused on classifying the images using only their visual features, while the text contained in an image brings a lot of information about the document. Even humans in certain situations cannot distinguish between two different types of documents only looking at them, without reading the text contained in them. Therefore it seems logical to find a way to combine both visual and text features of a document to classify it. Recently, researches in this field move in this new direction and R. Jain and C. Wigington in their work "Multimodal Document Image Classification" [1] explored various feature fusion techniques applied to the document classification task.

In this work, we want to prove the benefit of the main feature fusion method proposed by Jain and Wigington compared to pure image classifiers and pure text classifiers. Furthermore, we want to compare this method to the simpler approach to classify a document using the predictions from an image classifier and a text classifier, applying the so-called late fusion method.

In Section II we present the classification technique taken into consideration. In Section III we briefly describe the

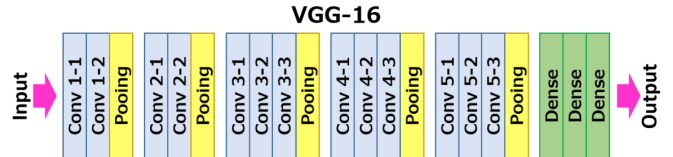


Fig. 1. The VGG16 architecture

dataset used for the experiments, which are exposed in Section IV. Finally, Section V contains some conclusions of ours experiments.

## II. STUDIED TECHNIQUES

### A. Image Classification

Classification using the images of the documents has been the state of the art in the document classification field until recently. This approach uses a CNN to extract visual features from the document.

Various network architectures have been used over the years, but we considered the VGG16 network, as it has proven to give the best performance in this task [3].

VGG16 is a model proposed in 2014 by K. Simonyan and A. Zisserman [4] and it is composed of 16 weight layers (13 convolutional and 3 FC layers), as shown in Figure X. When it was first proposed it achieved 92.7% accuracy on the ImageNet dataset, a dataset containing millions of images belonging to 1000 classes of objects from the real world, and since then it was intensively used in computer vision tasks.

Since VGG16 takes in input 224x224 images and considering its architecture, the whole network has over 130 millions of trainable parameters. This makes training from scratch computationally expensive, and it requires a huge domain-specific dataset.

It has become very popular applying transfer learning to CNN for computer vision tasks using networks pretrained on the ImageNet dataset. Even if document classification and object classification seem to be divergently domains, convolutional networks trained on the ImageNet dataset have proven to function as generalized feature extractors. In general, transfer learning has two main advantages:

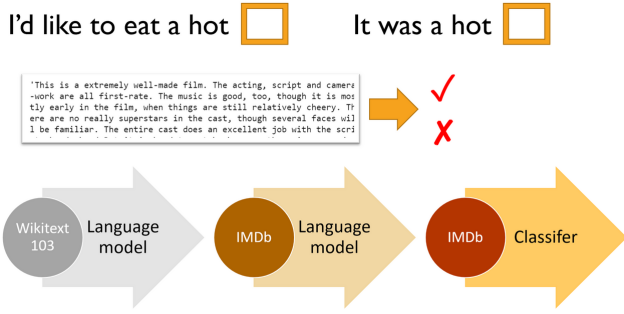


Fig. 2. The basic steps of ULMFiT applied to a sentiment analysis problem on the IMDb dataset: 1. Create/download a language model pre-trained on a large corpus (e.g. the Wikitext-103 dataset). 2. Fine-tune this language model using your target corpus. 3. Use the fine-tuned language model to train a classifier.

- allows to achieve good performance on small datasets, for which a training process from scratch would lead to overfitting
- speeds up the training process, requiring less epochs to get good results (often better than classical training approaches)

For document classification task, CNNs trained using transfer learning outperformed networks trained from scratch [3] [6].

### B. Text Classification

Text classification has been widely studied for application like spam filtering, sentiment analysis or for topic classification. There are many methods in recent studies that used a bag of words or character followed by a traditional classifier.

These approaches are competitive for many tasks when distinctive words or special characters are sufficient for classification. For example given a text that contains words like *To: From: ecc.* it might be very probable that the text comes from a mail.

Recent researches have explored word embeddings and deep neural networks models to capture a representation of text with a good success.

There are methods like Word2Vec or GLOVE that provides tools for a semantic classification, but for our purpose we have chose another type of approach.

We have used a linguistic approach called ULMFiT (*Universal Language Model Fine-Tuning*) [5], and in Figure 2 we can see its basic steps.

ULMFiT is an effective transfer learning method that can be applied to any task in NLP, and introduce techniques that are key for fine-tuning a language model and we use in our text classification for achieve large performance gains over the previous cited approaches.

The method is universal in the sense that it meets these practical criteria:

- Works across tasks varying in document size, number, and label type.
- Uses a single architecture and training process.
- Requires no custom feature engineering or pre-processing.

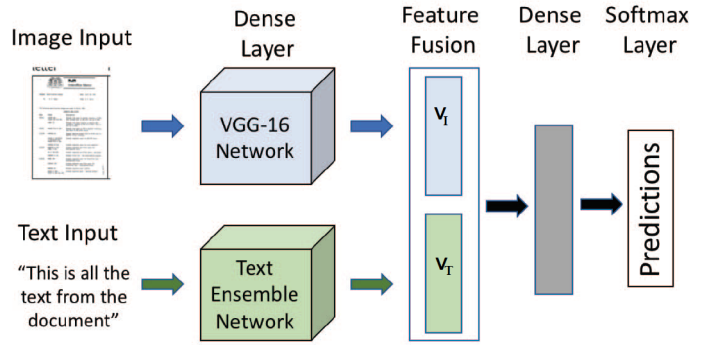


Fig. 3. A schematic representation of the feature fusion technique.

- Does not require additional in-domain documents or labels.

In fact in our experiments we have extract the text for the text classification from the image of RVL-CDIP sub-dataset that we have used for experiments, this extraction uses OCR and its explained in Section III.

### C. Multimodal Feature Fusion

The multimodal feature fusion approach aims to perform the classification combining the feature vector representations of both the visual ( $V_I$ ) and text ( $V_T$ ) modalities. This approach belongs to the so-called multimodal early fusion techniques, which combine the feature representations of the different modalities.

R. Jain and C. Wightington in [1] proposed a multimodal feature fusion method for document classification. The outputs from the second-last layer of both a VGG16 and a text classifier (i.e. excluding the softmax layers) are concatenated together and sent as input to a linear layer followed by a softmax to perform the classification. The architecture of the resulting network is shown in Figure 3.

### D. Multimodal Late Fusion

As a comparison for the performance of the multimodal feature fusion method, we also considered a multimodal late fusion technique. As opposed to early fusion ones, late fusion techniques combine the predictions of several classifiers from each modality.

In our case, we choose a very simple one: we concatenated the probabilities given by the last softmax layers of both the text and image classifiers and we gave them as input to a dense classifier, composed simply by a linear layer and a softmax.

## III. DATASET AND PREPROCESSING

We used the RVL-CDIP dataset for our experiments [10]. This collection contains 400.000 grayscale images in 16 classes, 25.000 images for each class. The dataset comes already split into 320.000 training images, 40.000 validation images and 40.000 test images. However, due to computational resources and time limitations, we created and used a mini-dataset containing only 4.000 images, preserving the proportions described above.

To train the VGG16 we resized all images to its expected input (224x224), ignoring the aspect ratios.

Concerning the text classifier, we created a textual dataset starting from the images. This textual dataset consists of a *csv* file with two columns: one containing the text read from an image and the second one containing the class label associated to that image. We used Tesseract OCR, one of the most famous open source OCR tools developed by Google, to read the text contained in each image.

We applied a preprocess step to the extracted text in which we removed the stop words. We used the list of stop words provided by NLTK, a toolkit used for NLP tasks. It is common to remove this stop words in NLP tasks so that more focus can be given to those words which contains more information about the meaning of the text.

Finally, we observed that sometimes the OCR system could not recognize any character in the image, such as for file folder documents where the image consists of a scan of the dossier, without any readable text. In these cases, we populated the text with a special character to let the text classifier work.

#### IV. EXPERIMENTS AND RESULT

In the following subsections, we are going to describe the performed experiments in detail. We first trained the image and the text classifier separately, then we used them to build and train the fusion models. Table I contains the results of the studied techniques on the test set.

##### A. Image Classification

For the image classifier, we used a VGG16 pretrained with the ImageNet weights and we replaced the fully connected part appropriately. We first trained only this last part from scratch, freezing the convolutional layers. We also used a heuristic technique to choose a good learning rate without having to run multiple training. We then applied a fine-tuning process, unfreezing the convolutional part and training the whole network. During the fine-tuning, we trained each layer with a different learning rate, greater for the last layers and lower for the first ones. It is common to follow this practice to do not make the network forget what it learned from the pre-training process once unfrozen all layers. Both training processes have been done using the 1-cycle policy, first presented by Leslie Smith [8] [9], for a maximum of 75 cycles, monitoring the validation accuracy and saving the best model. The VGG16 image classifier achieved 67.36% accuracy on our mini-dataset.

##### B. Text Classification

The text classifier training using ULMFiT consists of three phases. In the first one, we retrieved a language model pre-trained on the Wikitext-103 dataset [11]. It is a collection of over 100 million tokens extracted from the set of verified Good and Featured articles on Wikipedia, widely used for language modeling. Then we fine-tuned the language model using our textual dataset. Finally, we used the resulting language model to build and train an AWD-LSTM textual classifier.

TABLE I  
ACCURACY OBTAINED BY EACH METHOD ON OUR MINI-DATASET

METHOD	ACCURACY (%)
Image Class.	67.36
Text Class.	59.38
<b>Feature Fusion</b>	<b>76.38</b>
Late Fusion	74.30

The training process has been made with the same modalities used for the image classifier, and the trained textual classifier obtained 59.38% accuracy.

##### C. Multimodal Feature Fusion

We took the two trained models just described, we removed their last softmax layers and we used them to build the multimodal feature fusion classifier, as explained in Section II-C. In order to perform the training of this new network, we concatenated also the visual and the textual datasets, to send to the network a single tensor containing the image and the text extracted from that image and preprocessed. Once again, we trained from scratch only the last FC part keeping frozen the rest of the layers. Then we fine-tuned the whole network using a different learning rate for each layer. The accuracy gain was remarkable, obtaining 74.30%.

##### D. Multimodal Late Fusion

In the last experiment, we took our trained image and text classifiers as they were and we connected their output layer as described in Section II-D to build the multimodal late fusion model. This time made no sense to apply the fine-tuning process since in the late fusion approach we try to classify using only the predictions of classifiers from different modalities. Hence we only trained the new fully connected part, freezing the layers belonging to the image and text classifiers.

The multimodal late fusion model obtained 73.30% accuracy on the mini-dataset.

#### V. CONCLUSIONS

The feature fusion method proposed by Jain and Wigington in [1] is undoubtedly a sophisticated and smart technique. The authors analyzed the behaviour of the proposed methods on the images wrong classified by classifiers that used only the visual or the text features, and proved that the feature fusion classifier came over those difficult situations. An example can be seen in Figure V. Our experiments confirmed that the feature fusion allows to considerably improve the classification accuracy with respect to the classifiers using only the visual or the text features. Also the late fusion approach brings an important improvement compared to the two simple classifiers. However, with its simplicity cannot reach the feature fusion accuracy performance. It would be interesting to compare this two techniques on larger and different datasets.

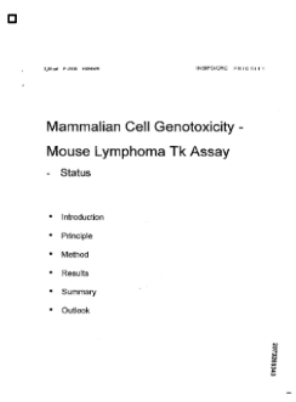
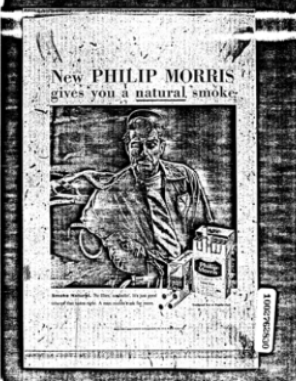
Document Image	OCR Text	Ground Truth	Predicted Classes
	2soppt P~et00 SRW INBIFOCRC PRIORITY Mammalian Cell Genotoxicity - Mouse Lymphoma Tk Assay - Status * Introduction ◦ Principle * Method ◦ Results * Summary * Outlook Evesszezoz	Scientific Report	Image: Presentation Text: Scientific Report Fusion: Scientific Report
	» ◦ So 5 R @ a o	Advertisement	Image: Advertisement Text: File Folder Fusion: Advertisement

Fig. 4. Examples of images from the RVL-CDIP where the image and text classifiers predicted wrong and the feature fusion model did well instead.

## REFERENCES

- [1] R. Jain and C. Wignington, "Multimodal Document Image Classification", 2019 International Conference on Document Analysis and Recognition (ICDAR), Sydney, Australia, 2019, pp. 71-77, doi: 10.1109/ICDAR.2019.00021.
- [2] The RVL-CDIP Dataset, Adam W. Harley, Alex Ufkes, and Konstantinos G. Derpanis, Url: <https://www.cs.cmu.edu/~aharley/rvl-cdip/>
- [3] Cutting the Error by Half: Investigation of Very Deep CNN and Advanced Training Strategies for Document Image Classification
- [4] Very Deep Convolutional Networks for Large-Scale Image Recognition, K. Simonyan and A. Zisserman
- [5] Universal Language Model Fine-tuning for Text Classification, Jeremy Howard fast.ai, University of San Francisco, j@fast.ai, Sebastian Ruder, Insight Centre, NUI Galway, Aylien Ltd., Dublin, sebastian@ruder.io
- [6] Document Image Classification with Intra-Domain Transfer Learning and Stacked Generalization of Deep Convolutional Neural Networks
- [7] A Comprehensive guide to Fine-tuning Deep Learning Models in Keras (Part I): <https://flyyufelix.github.io/2016/10/03/fine-tuning-in-keras-part1.html>
- [8] A DISCIPLINED APPROACH TO NEURAL NETWORK HYPERPARAMETERS: PART 1 – LEARNING RATE, BATCH SIZE, MOMENTUM, AND WEIGHT DECAY, Leslie N. Smith
- [9] <https://sgugger.github.io/the-1cycle-policy.html>
- [10] A. W. Harley, A. Ufkes, K. G. Derpanis, "Evaluation of Deep Convolutional Nets for Document Image Classification and Retrieval," in ICDAR, 2015
- [11] Stephen Merity, Caiming Xiong, James Bradbury, Richard Socher, "Pointer Sentinel Mixture Models"