Multimodal Classification of Document Embedded Images

Matheus Viana[1(✉)], Quoc-Bao Nguyen[2], John Smith[2], and Maria Gabrani[3]

[1] IBM Research Brazil, Rua Tutóia, 1157, São Paulo 04007-900, Brazil
mviana@br.ibm.com
[2] IBM Thomas J. Watson Research Center, 1101 Kitchawan Road, Route 134, Yorktown Heights, NY 10598, USA
[3] IBM Zurich Research Laboratory, Smerstrasse 4, 8803 Rschlikon, Switzerland

Abstract. Images embedded in documents carry extremely rich information that is vital in its content extraction and knowledge construction. Interpreting the information in diagrams, scanned tables and other types of images, enriches the underlying concepts, but requires a classifier that can recognize the huge variability of potential embedded image types and enable their relationship reconstruction. Here we tested different deep learning-based approaches for image classification on a dataset of 32K images extracted from documents and divided in 62 categories for which we obtain accuracy of $\sim 85\%$. We also investigate to what extent textual information improves classification performance when combined with visual features. The textual features were obtained either from text embedded in the images or image captions. Our findings suggest that textual information carry relevant information with respect to the image category and that multimodal classification provides up to 7% better accuracy than single data type classification.

1 Introduction

Images embedded in scientific, or other professional documents, such as financial reports, clinical trials report or internal company corpora, carry important visual information densely packed and structured. These images can represent many categories, such as diagrams, scanned tables, lesions in medical photos, scatter plots, and so on. Understanding the content of these images is crucial to answer questions that may not be possible by simply analyzing what is written in the document. For instance, one may want to know the average profit per year of a company based on a financial report that contains a table of profit per month. In another scenario, one may want to know whether there is a positive trend in a scatter plot. Obtaining the correct answer for these questions is a rather complex task in case the answer is not explicitly written in the main text. The first step towards a solution is to recognize the type of image one is dealing with. Next, the image is analyzed and its content is extracted in a category-dependent fashion. This process allows the construction of a knowledge graph for understanding concepts present in images and their relationships.

A. Fornés and B. Lamiroy (Eds.): GREC 2017, LNCS 11009, pp. 45–53, 2018.
https://doi.org/10.1007/978-3-030-02284-6_4

An important point is that, in most common applications, the image recognition is performed by an image classifier that relies on visual features to decide what is the most appropriated category. For instance, this is the case for most studies in document categorization and retrieval [6,9], functional decomposition [1] and modality detection of biomedical images [13].

However, many types of images also have relevant information in form of written text in addition to visual features. For instance, the word ocean may indicate that the image is a map, as well as the word chart may indicate a pie chart. This textual information can be extracted through OCR engines. A similar approach was used in [8,11,15], where the authors applied OCR to document images to extract textual features and to perform different tasks, such as document categorization and functional decomposition.

A less explored type of information corresponds to the captions of document images. In [2] the authors used images caption to perform the classification of images in three different classes. In this paper, we report the results of different experiments related to image classification. First, we tested different state-of-the-art network architectures to perform deep learning-based image classification. We also show that fusion techniques can improve the classification accuracy. Finally, we show that textual information extracted either from images via OCR engines or image captions carry important information and add significant level of accuracy per se.

To address the vast representation variability and extract the underlying structure we propose a multimodal deep learning approach which has been tested on two datasets described in Sect. 2. We present results in Sect. 3 and conclude the paper with discussion and summary in Sect. 4. In addition to testing different architectures for image classification.

The main contributions of our paper are *(i)* the systematic investigation of different networks architectures for classification of document images in a large number of classes; *(ii)* combination of both visual and textual features for classification of document images, where the textual features are obtained from OCR engines or images caption.

2 Methods and Datasets

2.1 Large Dataset

We generated a dataset of 32K images obtained from different sources, such as scientific papers, financial reports, medical reports and web search engines. The images were manually classified into 62 different categories that cover different domains such as *general*, *geological* and *molecular*. A few examples of images are shown in Fig. 1a. The goal was to use this dataset to investigate the performance of different deep learning-based methods in the task of document images classification.

The dataset was split into training (40%), validation (30%) and test (30%) for purposes of training convolutional neuronal networks (CNNs). We tested three CNN architectures, ResNet [7], Inception V3 [14] and Xception [3]. We

also performed fusion experiments by extracting features from the last layer of the CNNs and using them as input of SVM classifiers. We divide the fusion experiments into *average*, *stack* and *concat*, depending on how the features from two CNNs are combined.

We also used this dataset to test whether textual features extracted via OCR from the images could improve the performance of our classifiers. To do so, each image from our dataset was submitted to the Tesseract OCR engine to have their embedded text extracted. The resulting text of all the images were combined to generate a bag of words. We used the function spellcheck from the python library *textblob* to remove noisy words. We also used a snowball stemmer algorithm to reduce similar words to a common radical. Upon having this bag of words, each image was represented as binary vector where the i-th position is one if the i-th word of our bag does appear in the image. Finally, the binary vectors were used as input for a SVM classifier.

2.2 Small Dataset

We also generated a dataset of images extracted from papers from ArXiv. We downloaded all the 1425 documents that contained the word *seismic* in the abstract (as on Mar 15, 2018). We used PDFFigures [4] to extract metadata from these documents. PDFFigures is a tool that identifies tables and figures together with their corresponding captions. In total we extracted 3501 images and captions which have been manually classified into 6 classes, such as *plot*, *heat map* and *diagram*. Out of the total images extracted, we excluded 5.4% corresponding to text regions incorrectly extracted as images. Notice that this number is in agreement with the accuracy reported in [4]. Another 1.4% incorporated some part of the main text and had to have their bounding box manually adjusted. We did not use the tables extracted with PDFFigures and our goal was to use this small dataset to evaluate how the textual features extracted from the captions could improve the performance of our document images classifiers.

The PDF documents were converted into plain text files by using the *pdftotext* tool from the Poppler library (https://poppler.freedesktop.org/). All the text files have been pre-processed with same noise-removal and stemmer algorithms mentioned in the previous section and concatenated to create a representative corpora of our documents. The corpora contains $\sim 800K$ words and was used to train a *word2vec* algorithm with dimension 64 [12]. The trained *word2vec* model has a vocabulary of $\sim 30K$ words and was used to make inference over every image caption creating a numeric embedding representation of size 64 for each of these captions.

As mentioned above, the ground-truths classes of both large and small datasets were assigned by hand through visual inspection of each image.

3 Results

We used the large dataset described in Sect. 2.1 to test different deep learning-based approaches for classifying images extracted from documents. The results

are summarized in Table 1. We found that the fusion technique of extracting features from CNNs and using them as input of SVM classifier improves the baseline classification performance. Because transfer learning has been proven to be a successful approach for classifying document's images, [6], with exception of the networks marked with "*" in Table 1, all networks have been pre-trained on the ImageNet dataset. Only the last layer of those networks was fine-tuned on our dataset. The ResNet architectures marked with "*" have been trained from scratch and their number of layers was selected by optimizing the model accuracy in the range 50 to 101 layers.

Table 1. Classification performance of different architectures of neuronal networks on our testing set. Baseline represents the performance of a given network alone, while FE+SVM represented the performance obtained combining feature extraction from CNN and SVM classifier. Only best results are shown.

Model	Experiment	Input size	Acc. (%)
ReNet*101	baseline	256×256	76.79
ResNet*50	baseline	512×512	81.85
ReNet*101	FE+SVM	256×256	83.43
ReNet*50	FE+SVM	512×512	83.07
ReNet*58	baseline	128×128	84.63
ReNet*58	FE+SVM	128×128	84.98
ReNet*82	baseline	128×128	84.53
ReNet*82	FE+SVM	128×128	84.42
ReNet*(101+50)	FE+SVM (stack)	256×256 512×512	83.26
ReNet*(101+50)	FE+SVM (average)	256×256 512×512	84.88
ReNet*(101+50)	FE+SVM (concat)	256×256 512×512	84.41
InceptionV3	baseline	128×128	62.3
InceptionV3	baseline	256×256	78.87
Xception	baseline	256×256	75.21
Xception	baseline	512×512	81.03

Results in Table 1 indicates that the best performance in classifying images extracted from documents is achieved with input size 128×128. This is somehow intriguing given that larger images are capable of retaining fine grain image details. The best performance was achieved by using RestNet with 58 layers for extracting the features that feed to a SVM classifier. Also interesting that adding more layers (from 58 to 82) does not improve accuracy, which highlights the importance of selecting the appropriate number of layers.

We also used our large dataset to test whether the text embedded in the images together with the images themselves are capable of improving the classification accuracy. We found however that this type of data is very sparse as shown in Fig. 1b. In fact, less than 5% of our images displayed at least one word captured by the OCR engine.

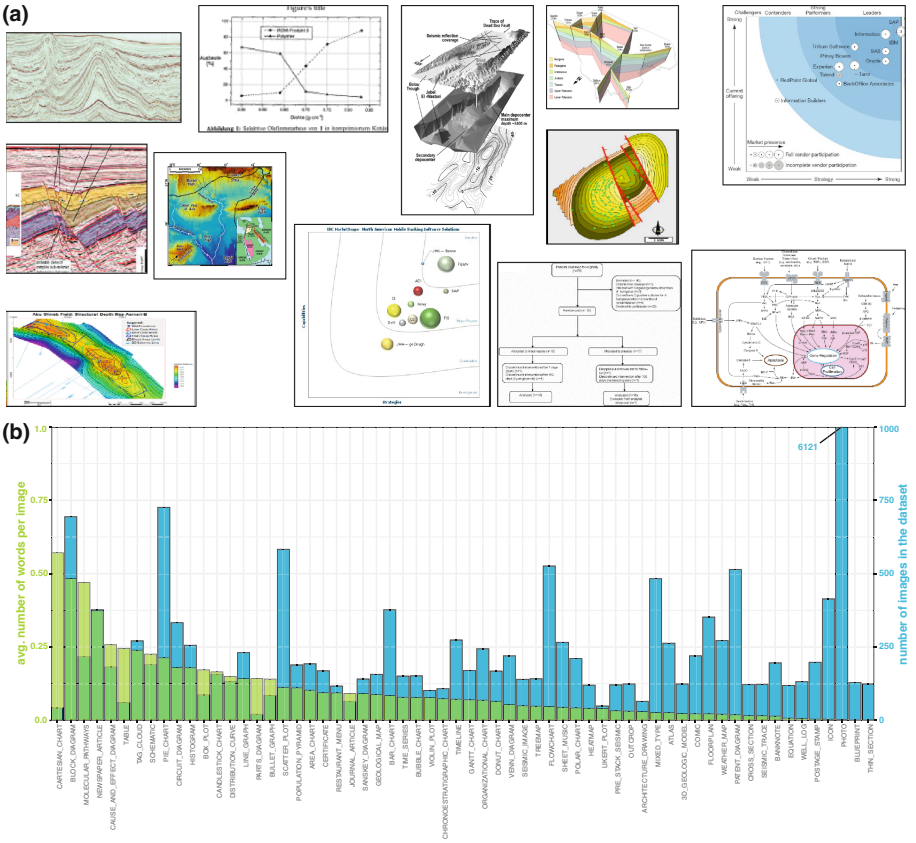

Fig. 1. (a) A few examples of images in our large dataset of 32K images. (b) Distribution of number of image in each class of our large dataset and average number of words fond by the OCR engine in each image for each class. The sparsity of the textual information in our dataset prevents us from using this type of data in a more effective way to improve image classification.

Despite the sparsity of this data, as a proof of concept that textual features embedded in images extracted from documents has some potential to positive affect the image classification performance, we created a subdataset in which we keep only the categories *block diagram* and *pie charts* for which we found a reasonable words to image ratio (∼0.37 words per image and 20% of the images

display at least one word). By using a four-layers multilayer perceptron (MLP; 256, 128, 64 and 2 units), we observed that textual features in isolation are capable of classifying the subdataset images with accuracy 8% above the baseline expected by chance. By inspecting the weights of the first layer of the MLP, we found the most relevant words are *start, input, program, director, information* and *diagram*, which seems to be words that often appear in images of class *block diagram*.

This result indicates that this data modality carries important information that could contribute to improve image classification. To test this hypothesis, we used the Xception network with input size of 128×128 to extract features from images of the subdataset and we combined these features with those generated by the 3rd layer of the MLP mentioned above. The concatenated features were used as input for a three-layers MLP (2048, 1024 and 2 units) responsible for classifying the images as *block diagram* or *pie chart*. We found that the combination of textual and visual features increases by 1% the accuracy in the testing set (acc. 0.92% vs 0.93%, respectively).

The previous result indicates that textual features carry additional information capable of improving classification accuracy when combined with visual features. To further explore the potential of multimodal classification, we used our small dataset described in Sect. 2.2 that combines visual features and textual features extracted from image captions. The textual features are encoded in a embedding representation created by using *word2vec* model. To check whether this type of representation carry any relevant information about the image category, we used *t-Stochastic Neighbor Embedding* (tSNE) [10] to reduce the embedding dimensionality from 64 to 2 as shown in Fig. 2a. The figure reveals some spatial structure, suggesting that image captions can be used for image classification.

Next, we tested the performance of a classifier based on textual features only. We used a four-layers MLP (256, 128, 64 and 6 units) to classify the captions embedding into one of the six classes. We split the data into training and testing sets (90% and 10%, respectively) and trained the model for 512 epochs. The maximum model accuracy on the testing set observed after 1024 epochs of training was 70.8%, which is significantly higher than what is expected by chance ($\sim$17%, see Fig. 2b). Although the experiments have been conducted in different datasets, this result suggests that figure captions have a lot more potential for classification of document figures than text embedded in the figures.

To check whether figure captions could contribute to improve image classification, we used the Xception network with input size of 128×128 to perform visual features extraction and we combined these features with the captions embedding. The concatenated features were used as input for a three-layers MLP (2048, 1024 and 2 units). As shown in Fig. 2c, we found the maximum accuracy in the testing set was about 7% greater than what is obtained when using visual features alone (79.6%).

To gain some insights of how the textual features combined with visual features are boosting the classification accuracy of document images, we inspected

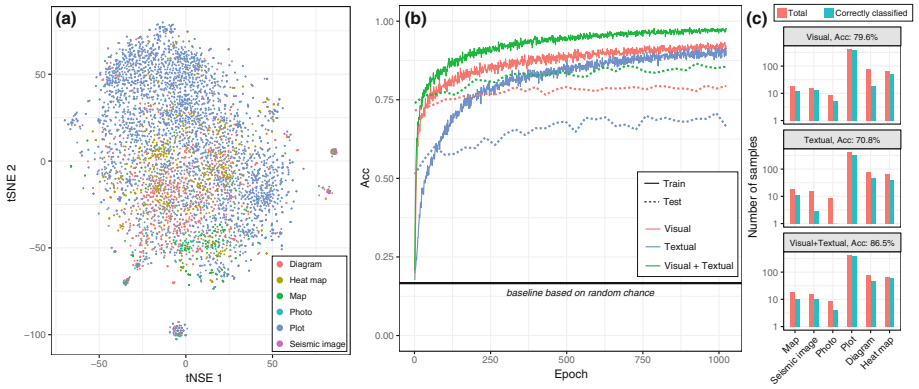

Fig. 2. (a) Embedding of document figure captions. A *word2vec* model of size 64 was trained on the corpora of all arXiv papers with the word *seismic* in the abstract. The embedding was created by applying word-by-word the trained model to every caption extracted out of those documents. We used *t-Stochastic Neighbor Embedding* (tSNE) [10] to reduce the embedding dimensionality from 64 to 2. (b) Accuracy for train and testing set as we train our single and multimodal models. (c) Classification performance of our best trained models for each class.

one sample (shown in Fig. 3 that was miss-classified by the Xception network that only uses visual features only. Without textual features, the image shown was classified as a *map*. Combining visual and textual features from its caption, the image was correctly classified as *diagram*. In Fig. 3 we also show the most relevant words of the corresponding caption that help in the correct image classification. We notice that words related to the concept of graph, such as *network, vertices, edge* and *connected*, display high importance for the correct classification, which is reasonable taking into account that the category *diagram* contains many examples of graphs and flowcharts.

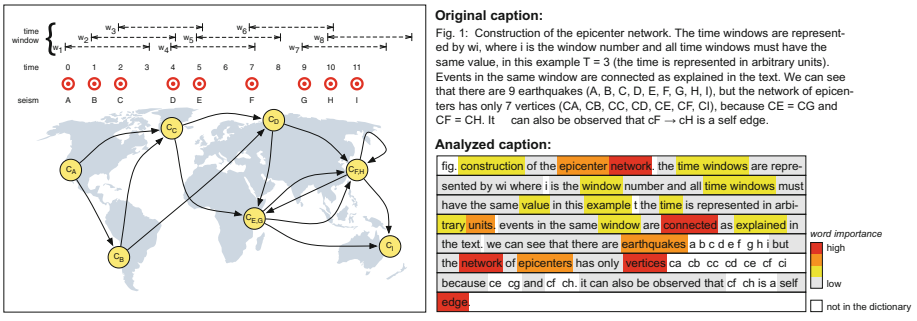

Fig. 3. Figure extracted from [5] that was correctly classified as *diagram* when using visual and textual features together. We also show the impact each word had for the image classification.

4 Conclusion

Image understanding is a crucial part in the process of document-based knowledge construction. As part of this step, image classification is of fundamental importance so that appropriate algorithms can be applied to specific types of images for content extraction. Here we used a large dataset to show how feature extraction combined with SVM classifier can improve the process of image classification. We also investigate the role of textual features in boosting image classification when combined with visual features. First we show how text extracted from images through OCR engines can be used as an additional source of information for image classification. Although some improvement was observed, we found this feature to be very spare and its use require appropriate high-resolution datasets for which OCR engines are reliable and for which the text extraction make sense. Finally, we used a small dataset that contains images extracted from documents and their captions to further explore the combination of visual and textual features. A corpora was created and used to train a word2vec model. The model was used for creating an embedding representation for each caption. By means of tSNE 2D projection we observed that the embedding displays spatial structure in terms of images categories. We found a significant performance improvement ($\sim$7%) in image classification when visual features extracted via ConvNet were combined with textual features represented by captions embedding. Our results also suggest that the use of textual information is more effective for particular classes of images, such as *diagrams* and less effective for other classes, such as *photo* (see Fig. 2). In future experiments we would like to test different approached for weighted multimodal classification, i.e. classification of images based on different types of data such data each data has a different weight depending on the image class. These weights will be embedded in the deep learning algorithm such that the whole system can be trained in a end-to-end fashion.

References

1. Chao, H., Fan, J.: Layout and content extraction for PDF documents. In: Marinai, S., Dengel, A.R. (eds.) DAS 2004. LNCS, vol. 3163, pp. 213–224. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-28640-0_20
2. Cheng, B., Stanley, R.J., Antani, S., Thoma, G.R.: Graphical figure classification using data fusion for integrating text and image features. In: 2013 12th International Conference on Document Analysis and Recognition (ICDAR), pp. 693–697. IEEE (2013)
3. Chollet, F.: Xception: Deep learning with depthwise separable convolutions. arXiv preprint (2016)
4. Clark, C.A., Divvala, S.K.: Looking beyond text: extracting figures, tables and captions from computer science papers. In: AAAI Workshop: Scholarly Big Data (2015)
5. Ferreira, D.S., Ribeiro, J., Papa, A.R., Menezes, R.: Towards evidences of long-range correlations in seismic activity. arXiv preprint arXiv:1405.0307 (2014)

6. Harley, A.W., Ufkes, A., Derpanis, K.G.: Evaluation of deep convolutional nets for document image classification and retrieval. In: 2015 13th International Conference on Document Analysis and Recognition (ICDAR), pp. 991–995. IEEE (2015)
7. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
8. Ittner, D.J., Lewis, D.D., Ahn, D.D.: Text categorization of low quality images. In: Symposium on Document Analysis and Information Retrieval, pp. 301–315. Citeseer (1995)
9. Kang, L., Kumar, J., Ye, P., Li, Y., Doermann, D.: Convolutional neural networks for document image classification. In: 2014 22nd International Conference on Pattern Recognition (ICPR), pp. 3168–3172. IEEE (2014)
10. Maaten, L.V.D., Hinton, G.: Visualizing data using T-SNE. J. Mach. Learn. Res. **9**, 2579–2605 (2008)
11. Maderlechner, G., Suda, P., Brückner, T.: Classification of documents by form and content. Pattern Recognit. Lett. **18**(11–13), 1225–1231 (1997)
12. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems, pp. 3111–3119 (2013)
13. Miranda, E., Aryuni, M., Irwansyah, E.: A survey of medical image classification techniques. In: International Conference on Information Management and Technology (ICIMTech), pp. 56–61. IEEE (2016)
14. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2818–2826 (2016)
15. Taylor, S.L., Lipshutz, M., Nilson, R.W.: Classification and functional decomposition of business documents. In: Proceedings of the Third International Conference on Document Analysis and Recognition, vol. 2, pp. 563–566. IEEE (1995)