

# Dynamic Lexicon Generation for Natural Scene Images

Yash Patel<sup>1,2</sup>, Lluis Gomez<sup>2</sup>, Marçal Rusiñol<sup>2</sup>, and Dimosthenis Karatzas<sup>2</sup>

<sup>1</sup> CVIT, IIIT Hyderabad, India.

[yash.patel@students.iiit.ac.in](mailto:yash.patel@students.iiit.ac.in)

<sup>2</sup> Computer Vision Center, Universitat Autònoma de Barcelona.

[{lgomez,marcal,dimos}@cvc.uab.es](mailto:{lgomez,marcal,dimos}@cvc.uab.es)

**Abstract.** Many scene text understanding methods approach the end-to-end recognition problem from a word-spotting perspective and take huge benefit from using small per-image lexicons. Such customized lexicons are normally assumed as given and their source is rarely discussed. In this paper we propose a method that generates contextualized lexicons for scene images using only visual information. For this, we exploit the correlation between visual and textual information in a dataset consisting of images and textual content associated with them. Using the topic modeling framework to discover a set of latent topics in such a dataset allows us to re-rank a fixed dictionary in a way that prioritizes the words that are more likely to appear in a given image. Moreover, we train a CNN that is able to reproduce those word rankings but using only the image raw pixels as input. We demonstrate that the quality of the automatically obtained custom lexicons is superior to a generic frequency-based baseline.

**Keywords:** scene text, photo OCR, scene understanding, lexicon generation, topic modeling, CNN

## 1 Introduction

Reading systems for text understanding in the wild have shown a remarkable increase in performance over the past five years [1, 2]. However, the problem is still far from being considered solved with the best reported methods achieving end-to-end recognition performances of 87% in focused text scenarios [3, 4] and 53% in the more difficult problem of incidental text [5].

The best performing end-to-end scene text understanding methodologies address the problem from a word spotting perspective and take a huge benefit from using customized lexicons. The size and quality of these custom lexicons has been shown to have a strong effect in the recognition performance [6].

The source of such per-image customized lexicons is rarely discussed. In most academic settings such custom lexicons are artificially created and provided to the algorithm as a form of predefined word queries. But, in real life scenarios lexicons need to be dynamically constructed.

In one of the few examples in literature, Wang et. al. [7] used Google’s “search nearby” functionality to built custom lexicons of businesses that might appear in Google Street View images. In the document analysis domain, different techniques for adapting the language models to take into account the context of the document have been used, such as language model adaptation [8] and full-book recognition techniques [9]. Such approaches are nevertheless only feasible on relatively large corpuses were word statistics can be effectively calculated and are not applicable to scene images where text is scarce.

On the other hand, scene images contain rich visual information that could provide the missing context to improve text detection and recognition results. In the view of the authors, reading text in the wild calls for holistic scene understanding in a way where visual and textual cues are treated together providing mutual feedback for each other’s interpretation.

In this paper, we take a first step in this direction, and we propose a method that generates contextualized lexicons based on visual information. For this we make the following contributions: first, we learn a topic model using Latent Dirichlet Allocation (LDA) [10] using as a corpus textual information associated with scene images combined with scene text. This topic model is suitable for generating contextualized lexicons of scene text given image descriptions. Subsequently, we train a deep CNN model, based on the topic model, that is capable to produce on its output a probability distribution over the topics discovered by the LDA analysis directly from the image input. This way our method is able to generate contextualized lexicons for new (unseen) images directly from their raw pixels, without the need of any associated textual content. Moreover, we demonstrate that the quality of such automatically obtained custom lexicons is superior to generic frequency-based lexicons in predicting the words that are more likely to appear as scene text instances in a given image.

## 2 Related Work

End-to-end scene text recognition pipelines are usually based in a multi-stage approach, first applying a text detection algorithm to the input image and then recognizing the text present in the cropped bounding boxes provided by the detector [11].

Scene text recognition from pre-segmented text has been approached in two different conditions: using a small provided lexicon per image (also known as the word spotting task), or performing unconstrained text recognition, i.e. allowing the recognition of out-of-dictionary words.

Many of the existing text recognition methods [7, 6, 12–14] rely on individual character segmentation and recognition. After that, character candidates are grouped into larger sequences (words and text lines) using spatial and lexicon-based constraints. Such methods differ, apart from the features and classifiers used for individual character classification, in their language models, and the inference methods used to find the best character sequence, e.g. pictorial structures [7], Conditional Random Fields (CRF) [15], Viterbi decoding [6, 12, 13], or

Beam Search [14]. Language models are usually based on a dictionary of the most frequent words in a given language and a character n-gram (usually a bi-gram). A much stronger language model, relying on large-scale data center infrastructure, is used in [14] combining a compact character-level 8-gram model and a word-level 4-gram model.

State of the art language models for document-based OCR have demonstrated good performance in scene text recognition when text instances can be properly binarized [16, 17].

In the case of the word spotting and retrieval tasks it is also possible to make use of holistic word recognizers that perform recognition without any explicit character segmentation (Goel 2013, Almazan 2013, jaderberg2016reading).

Obviously, either language-model based approaches and holistic word recognizers may benefit from using per-image customized lexicons: by reducing their search space or (in the case of holistic recognizers) the number of possible class-labels. As an example, table 1 shows end-to-end recognition performance in [6] for different sizes of the per-image lexicon.

	5 distractors	20 distractors	50 distractors	860 distractors	Open vocabulary
F-score	76%	74%	72%	67%	38%

Table 1: Recognition performance drops when adding distraction words in the lexicon [6].

However, having a small lexicon containing all the words that may appear in a given image is not realistic in many cases. Even for methods using large (frequency based) lexicons (e.g. the 90k word dictionary used in [4]), it is not possible to recognize out-of-dictionary words such as telephone numbers, prices, url's, email addresses, or to some extent product brands. Notice that in the last edition of the ICDAR Robust Reading Competition [2] out-of-dictionary words are not included in the customized dictionaries provided for end-to-end recognition, as it is not realistic that such queries would be available in a real-life scenario.

An interesting method for reducing an initial large lexicon to a small image-specific lexicon is proposed in [18]. Since having a large lexicon poses a problem for CRF based methods because pairwise potentials become too generic, they propose a lexicon reduction process that alternates between recomputing priors and refining the lexicon.

Wang et. al. [7] propose the use of geo-localization information to build custom lexicons of businesses that might appear in Google Street View images. This multimodal approach to generate contextualized lexicons from GPS data clearly helps in recognition of out-of-dictionary words that are strongly correlated with the location from which an image is taken, e.g. street names, touristic attractions, business front stores, etc.

In this paper we propose a method that generates contextualized lexicons based only on visual information. The main intuition of our method is that visual information may provide in some cases a valuable cue for text recognition algorithms: there are some words for which occurrence in a natural scene image correlates directly with objects appearing in the image or with the scene category itself. For example, if there is a telephone booth in the image the word “telephone” has a large probability of appearance, while if the image is a mountain landscape the “telephone” word is less likely to appear.

Evidence of this correlation between visual and textual information in natural scene images has been recently reported by Movshovitz et al. in [19]. A Deep Convolutional Neural Network trained for fine-grained classification of storefront street view images implicitly learns to read, i.e. to use textual information, when needed, despite it has been trained without any annotated text or language models. The network, by learning the correct representation for the task at hand, learned that some words are correlated with specific types of businesses, up to a point in which if the text in correctly classified images is removed the net loses classification confidence about their correct class. Moreover, the network is able to produce relevant responses when presented with a synthetic image containing only textual information (a word) that relates with a specific business.

On a totally different application but also in relation with exploiting the correlation between visual and textual information, Feng et al. [20] proposes a method that uses the topic modeling framework for generating automatic image annotations and text illustration. They presented a probabilistic model based on the assumption that images and their co-occurring textual data are generated by mixtures of latent topics.

A topic model [21, 10] is a type of statistical model for discovering the abstract “topics” that occur in a collection of text documents. Topic modeling has been applied successfully in many text-analysis tasks such as document modeling, document classification, semantic visualization / organization, and collaborative filtering. But they also have applications in Computer Vision research. They have been used for unsupervised image classification by Li et al. [22] when Bag-of-Visual-Words was a dominant method for image classification. Nowadays they are a common tool for automatic image annotation methods.

In this paper we present a method that generates contextualized lexicons based on visual information. We use a similar approach to [20] in topic models to exploit correlation between visual and textual information. But distinctly to [20] we do not aim at generating annotations of the image, but instead the words that are more likely to appear in the image as scene text instances.

Our method is also related with the work of Zhang et al. [23] in which we use Latent Dirichlet Allocation (LDA) [10] within the topic modeling framework to supervise the training of a deep neural network (DNN), so that DNN can approximate the LDA inference. An idea that is motivated by the transfer learning approach of [24]. However, in our method we train a deep CNN that takes images as input instead of a text Bag-of-Words as in [23].

### 3 Method

The underlying idea of our lexicon generation method is that the topic modeling statistical framework can be used to predict a ranking of the most probable words that may appear in a given image. For this we propose a three-fold method: First, we learn a LDA topic model on a text corpus associated with the image dataset. Second, we train a deep CNN model to generate LDA's topic-probabilities directly from the image pixels. Third, we use the generated topic-probabilities, either from the LDA model (using textual information) or from the CNN (using image pixels), along with the word-probabilities from the learned LDA model to re-rank the words of a given dictionary. Figure 1 shows a diagram representation of the overall framework.

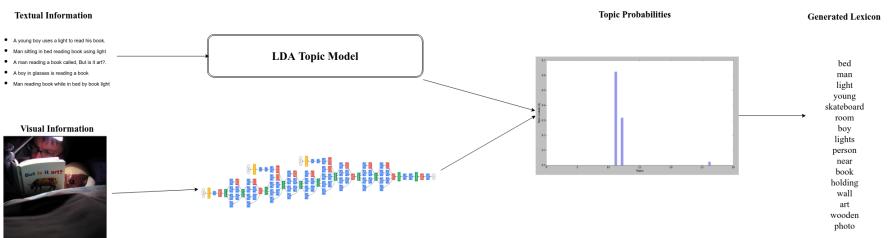


Fig. 1: We learn a LDA topic model on a text corpus associated with scene images. Then we train a deep CNN model to generate the probabilities over latent topics directly from image pixel information.

The training samples in our framework are composed by a couple of visual information (an image) and some associated textual information (e.g. a set of image captions and/or the annotations of words that appear in the image), see Figure 2 (a). Our model assumes that the textual information describe the content of the image either directly or indirectly and hence can be used to generate contextualized lexicons for natural images.

In the next section we explain how we learn the LDA topic model to discover latent topics from training data by using only the textual information. Then in section 3.2 we show how it is possible to train a deep CNN model to predict the same probability distributions over topics as the LDA model but using only the image pixels (visual information) as input. Finally, in section 3.3 we explain how using the topic probabilities we can generate word rankings, i.e. a per-image ranked lexicon, for new (unseen) images.

#### 3.1 Learning the LDA topic model using Textual Information

Our method assumes that the textual information associated with the images in our dataset is generated by a mixture of latent topics. Similar to [20], we propose

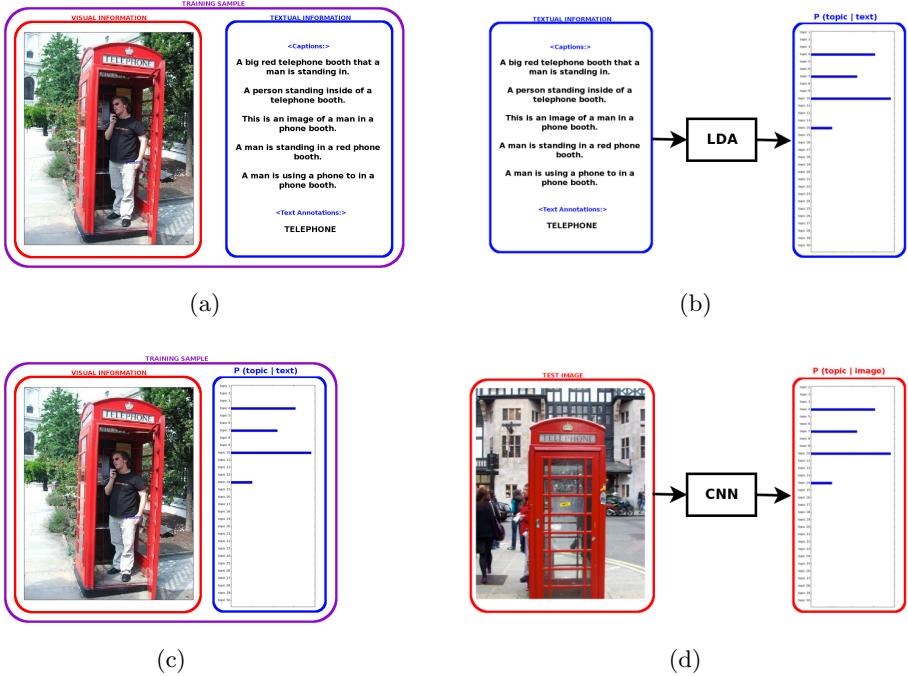


Fig. 2: (a) Our training samples consist in a couple of an image and some associated textual content. (b) Using a topic model we can represent the textual information as a probability distribution over topics  $P(\text{topic} \mid \text{text})$ . (c) Training samples for our CNN use those probability values as labels. (d) The CNN takes an image as input and produces on its output a probability distribution over topics  $P(\text{topic} \mid \text{image})$ .

the use of Latent Dirichlet Allocation [10] for discovering the latent topics in the dataset's text corpus, and thus to represent the textual information associated with a given image as a probability distribution over the set of discovered topics.

As presented in [10], LDA is a generative statistical model of a corpus (a set of text documents) where each document can be viewed as a mixture of various topics, and each topic is characterized by a probability distribution over words. LDA can be represented as a three level hierarchical Bayesian model. Given a text corpus consisting of  $M$  documents and a dictionary with  $N$  words, Blei et al. define the generative process [10] for a document  $d$  as follows :

- Choose  $\theta \sim Dir(\alpha)$ .
- For each of the  $N$  words  $w_n$ :
  - Choose a topic  $z_n \sim Multinomial(\theta)$ .
  - Choose a word  $w_n$  from  $p(w_n \mid z_n, \beta)$ , a multinomial probability conditioned on the topic  $z_n$ .

where  $\theta$  is the mixing proportion and is drawn from a Dirichlet prior with parameter  $\alpha$ . As [10] suggests,  $\alpha$  and  $\beta$  are assumed to be sampled once in the process of generating a corpus. This way the documents are represented as topic probabilities  $z_{1:K}$  (being  $K$  the number of topics) and word probabilities over topics. The learned LDA model has two sets of parameters, the topic probabilities given documents  $P(z_{1:K} | d)$  and the word probabilities given topics  $P(w | z_{1:K})$ . This way any new (unseen) document can be represented in terms of a probability distribution over topics of the learned LDA model by projecting it in the topic space.

Notice that in our framework a document corresponds to the textual information associated to an image (e.g. image captions and scene text annotations). Thus, the text corpus is the set of all textual information (documents) in the whole dataset. By learning the LDA topic model using this corpus we discover a set of latent topics in our dataset, and we can represent the textual information associated to a given image as a probability distribution over those topics  $p(topic|text)$  as shown in Figure 2 (b).

### 3.2 Training a CNN to predict probability distributions over LDA's topics

Once we have the LDA topic model, we want to train a deep CNN model to predict the same probability distributions over topics as the LDA model does for textual information, but using only the raw pixels of new unseen images.

For this we can generate a set of training (and validation) samples as follows: given an image from the training set we represent its corresponding textual information (captions) as probability values over the LDA's topics. These probability values are used as labels for the given image as shown in Figure 2 (c).

This way we obtain a set of  $M$  training (and validation) examples of the form  $\{(x_1, y_1), \dots, (x_M, y_M)\}$  such that  $x_i$  is an image and  $y_i$  is the probability distribution over topics obtained by projecting its associated textual information into the LDA topic space.

Using this training set we train a deep CNN to predict the probability distribution  $y_i$  for unseen images, see Figure 2 (d), directly from the image pixels. In fact, we use a transfer learning approach here in order to shortcut the training process by fine-tuning the well known Inception [25] deep CNN model. Details on the training procedure are given in section 4.2.

### 3.3 Using topic models for generating word ranks

Once the LDA topic model is learned as explained in section-3.1, we can represent the textual information corresponding to an unobserved image as probability distribution over the topics of LDA model  $P(topic | text)$ , which is done by projecting the textual information to the topic-space. Since the contribution of each word to each topic,  $P(word | topic)$  was pre-computed when we learned the LDA model, we can calculate the probability of occurrence for each word in the dictionary  $P(word | text)$  as follows:

$$P(word | text) = \sum_{i=1:K} (P(word | topic_i)P(topic_i | text)) \quad (1)$$

Similarly, once the deep CNN is trained as explained in section-3.2, we can obtain the probability distribution over topics for an unseen image  $P(topic | image)$  as the output of the CNN when feeding the image pixels on its input. Again, since the word-probability for each topic which  $P(word | topic)$  is known from the corresponding LDA model, which we used to supervise the training of deep CNN's training, we can calculate the probability of occurrence of each word in the dictionary  $P(word | image)$  as follows :

$$P(word | image) = \sum_{i=1:K} (P(word | topic_i)P(topic_i | image)) \quad (2)$$

Using the obtained probability distributions over words (i.e.  $P(word | text)$ ), or  $P(word | image)$ ) we are able to rank a given dictionary in order to prioritize the words that have more chances to appear in a given image.

In the following section we show how the the word rankings obtained from both approaches are very similar, which demonstrates the capability of the deep CNN to generate topic probabilities directly from the image pixels. Moreover, the rankings generated this way prove to be better than a frequency-based word ranking in predicting which are the expected scene text instances (words) to be found in a given image.

## 4 Experiments and Results

In this section we present the experimental evaluation of the proposed method on its ability to generate lexicons that can be used to improve the performance of systems for reading text in natural scene images. First, we present the datasets used for training and evaluation in section 4.1. Then, in section 4.2, we provide the implementation details of our experiments. In section 4.3, we analyze the performance of the word rankings obtained by representing image captions as a mixture of LDA topics as detailed in section 3.3. Finally in section 4.4, we show the performance of the word ranking obtained with our CNN network trained for predicting topic probabilities.

### 4.1 Datasets

In our experiments we make use of two standard datasets, namely the MS-COCO [26] and the COCO-Text [27] datasets.

The MS-COCO is a large scale dataset providing task-specific annotations for object detection, segmentation, and image captioning. The dataset consists of 2.5 million labeled object instances among 80 categories in 328K images of complex everyday scenes. Images are annotated with multiple object instances and with 5 captions per image.

COCO-Text is a dataset for text detection and recognition in natural scene images that is based on the MS-COCO dataset. The images in this dataset were not taken with text in mind and thus it contains a broad variety of text instances. The dataset consists of 63,686 images, 173,589 text instances (words) and 3-fine grained text attributes. The dataset is divided in 43,686 training images and 20,000 validation images.

COCO-Text images are a subset of MS-COCO images, thus for our experiments we use the ground truth information of both datasets: image captions from MS-COCO and text instances (word transcriptions) from COCO-Text.

Since both the training and validation images of COCO-Text are a subset of the MS-COCO training set, we have done the following partition of the data: for training purposes we use the training and validation sets of MS-COCO but removing the images that are part of the validation set in COCO-Text. For validation purpose we use the validation set of COCO-Text. This way our training set consists of 103287 images and our validation set of 20000 images.

Apart of the MS-COCO and COCO-Text datasets we have used the entire english-wikipedia text, consisting of around 4 million text documents, for computing the word-frequency based lexicon used as a Baseline for word-rankings evaluation.

## 4.2 Implementation Details

In our experiments involving topic modeling we have used the gensim [28] Python library for learning and inferring the LDA model. We have learned multiple LDA models with a varying number of topics and have compared word ranking results as shown in section 4.3.

On the other hand, we have used the TensorFlow [29] framework for fine-tuning of the Inception-v3 model [25]. We have trained the final layer of the net from scratch, accommodating it to the size of our topic modeling task, and leaving the rest of the net untouched. We used the cross entropy loss function and Gradient Descent optimizer with a fixed learning rate of 0.01 and a batch size of 100 for 100k iterations.

## 4.3 LDA word rankings from image captions

In this section we evaluate the performance of the different word rankings obtained with our method on predicting text instances (words) for COCO-Text validation images. The setup of the experiment is as follows:

**Corpus:** We learned the LDA topic model using two different corpuses: (1) we use 63686 documents made using the word annotations from both the train and validation images of COCO-Text and their corresponding captions (from MS-COCO); (2) we do the same but using only the 43686 images in the train set of COCO-Text and their corresponding captions (from MS-COCO).

**Dictionary:** We do experiments with two different dictionaries: (1) The list of 33563 unique annotated text instances (words) in the COCO-Text dataset; and

(2) a generic dictionary of approximately 88172 words used in [4], but removing stop words thus giving rise to a dictionary of 88036 words.

**Word-rankings using LDA:** For each image, the words of the dictionary are ranked by obtaining the word probabilities as mentioned in section 3.3.

**Baseline ranking:** Dictionary words are ranked according to their frequency of occurrences. Frequency of occurrence of dictionary words is computed on wikipedia-english corpus.

Given a fixed dictionary we are interested in word rankings that are able to prioritize the words that are more likely to appear in a given image as scene text instances. Thus, we propose the following procedure to evaluate and compare different word rankings: for every word ranking we count the percentage of COCO-Text ground truth (validation) instances that are found among the top- $N$  words of the re-ranked dictionary. This way we can plot curves illustrating the number of COCO-Text instances found in different word lists (lexicons) that correspond to certain top percentages of the ranked dictionary. The larger the area under those curves the better is a given ranking.

Figure 3 evaluate our method for a varying number of topics of the LDA model, and compare their performance with the baseline frequency-based ranking using the above mentioned text corpuses and dictionaries. The x-axis in the plots represents percentage of words of the re-ranked dictionary. The y-axis represents percentage of instances of the validation set of COCO-Text dataset in the top- $N$  percent of the re-ranked dictionary.

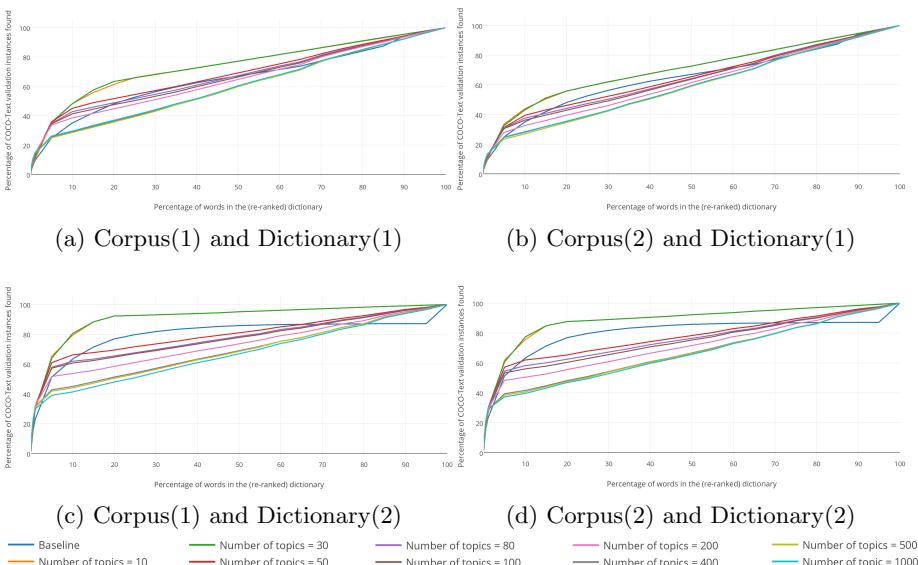


Fig. 3: Word ranking performance comparison.

Notice that we analyze our method only for the COCO-Text word instances that are present in the dictionary, this is why using 100% of dictionary words we always reach 100% of COCO-Text instances.

As can be appreciated the number of topics in the LDA topic model is an important parameter of the method. The best performance for our automatically generated rankings are found for the 30 topics model. In such a case the performance of the LDA based rankings is superior to the baseline in all the experiments. This demonstrates that the topic modeling analysis we propose is able to predict the occurrence of words as scene text instances much better than a frequency-based dictionary.

Figure 4 shows the performance comparison of the rankings generated with the 30 topics LDA model in both dictionaries. Obviously using the textual content associated with validation images, in Corpus(1), for learning the LDA topic model provides an extra boost to the method’s performance. Still the word rankings provided by the LDA model learned only from training data, Corpus(2), clearly outperform the baseline ranking.

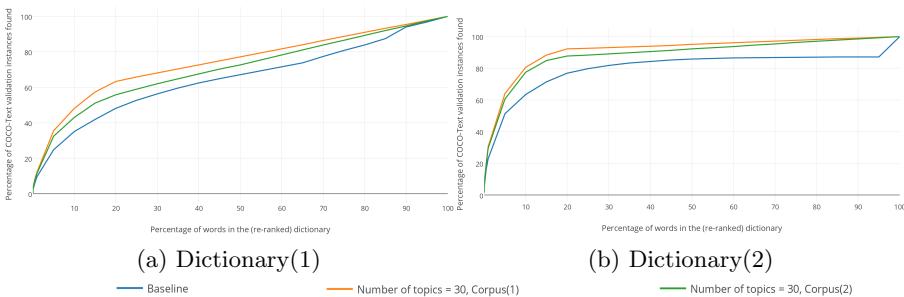


Fig. 4: Word ranking performance comparison by using Corpus(1) and Corpus(2) with each dictionaries.

#### 4.4 CNN word rankings

In this experiment we evaluate the performance of the deep CNN network trained with the procedure detailed in sections 3.2 and 4.2. Figure 5 shows the performance comparison of the word rankings obtained by the LDA model using 30 topics as in the previous section, and the word rankings obtained with the CNN as explained in section 3.3. It is important to notice here that for training the CNN the train images’ labels are generated from the LDA model learned only with Corpus(2). This is, our CNN model has never seen validation data (neither images or textual content) in a direct or indirect way.

As can be seen the CNN is able to produce word rankings with almost the same performance as projecting the images’ captions in the LDA space, but using

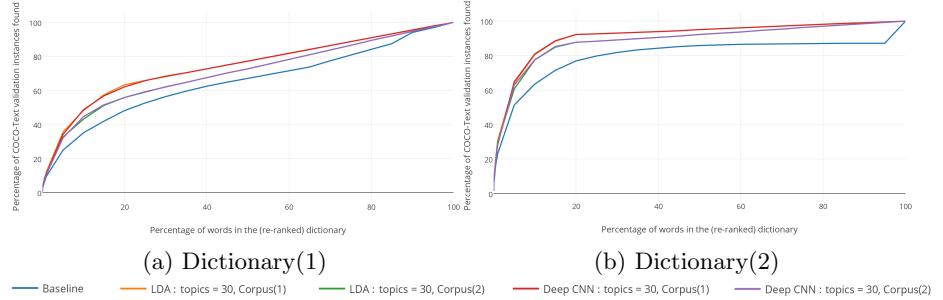


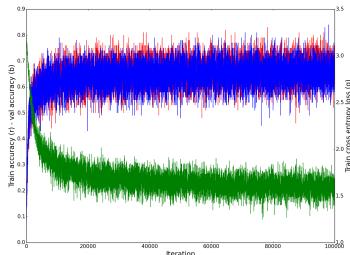
Fig. 5: Word ranking performance comparison of LDA model and deep CNN with 30 topics for Dictionaries (1) and (2).

only the image raw pixels as input. Using the CNN for predicting the probability distribution over 30 topics for a given image takes takes 54ms.

Figure 6 shows the cross entropy loss of the CNN during the training process, up to 100k iterations. We also show the top-1 topic classification accuracy (i.e. as when we evaluate a classification task) because it's efficient and gives us a rough estimation on how the network is performing at every iteration. Thus, for visualization we calculate accuracy by looking only at the most important topic in ground-truth labels and in the CNN predictions.

Once the CNN model is trained we analyze its performance more precisely by looking at the top-N most important topics, defined as the set of top-N topics for which the sum of their probabilities reaches a certain threshold. Figure 6 shows the CNN precision at  $N$  ( $P@N$ ) calculated this way for the train and validation sets. Notice that  $N$  might change for each image.

As can be appreciated in the table the CNN model is able to approximate the learned topic model consistently in both training and validation sets. While the obtained precision at  $N$  are far from a perfect model, we can see that in average the CNN is able to predict the top-2 topics pretty well in nearly 70%



Probability	Avg. N	Train P@N	Val. P@N
0.4	1.1	67.35%	66.61%
0.5	1.2	69.26%	68.59%
0.6	1.4	70.64%	70.06%
0.7	1.7	70.23%	69.78%
0.8	2	67.92%	67.56%
0.9	2.4	62.72%	62.17%

Fig. 6: Cross entropy loss and top-1 classification accuracy during the CNN training process up to 100k iterations (left). CNN model train and validation precision at  $N$  for different probabilities.

of the images. Moreover, since as shown in Figure 5 the performance of the word rankings obtained directly from the topic model and the CNN are almost identical, we can conclude that these values are only an estimator of the CNN real performance. In other words, the word rankings produced by the CNN can be as good as the ones using the LDA topic models even if the CNN prediction is not 100% accurate.

Figure 7 shows qualitative results in which it can be appreciated the effectiveness of the proposed method to produce word rankings that prioritize the text instances annotated in different sample images. Figure 8 shows some unsuccessful cases.

	<b>Word annotations :</b> <b>Word Rank</b>  Fire : 2 Hydrant : 4		<b>Word annotations :</b> <b>Word Rank</b>  wii : 11
	<b>Word annotations :</b> <b>Word Rank</b>  high : 182 street : 1		<b>Word annotations :</b> <b>Word Rank</b>  TENNIS : 1

Fig. 7: Qualitative successful results of generated word rankings.

Images in Figures 7 and 8 have been selected from the validation set in order to show a diversity of cases in which the proposed method produces particularly interesting results that can be potentially leveraged by end-to-end reading systems. For example, for the bottom-right image in Figure 7 the top ranked word in the 33K words dictionary(1) is “TENNIS”, a word instance that is partially occluded in the image and whose recognition would be very difficult without the context provided by the scene.

## 5 Conclusion

In this paper we have presented a method that generates contextualized per-image lexicons based on visual information. This way we take a first step towards

	<b>Word annotations :</b> <b>Word Rank</b> florida : 7919 time : 167 <b>Top Ranked words</b> clock : 127 red : 5 front : 13 restaurant : 213		<b>Word annotations :</b> <b>Word Rank</b> midnight : 13900 exit : 2041 <b>Top Ranked words</b> sign : 2 building : 3 street : 4 road : 29
	<b>Word annotations :</b> <b>Word Rank</b> oracle : 8488 <b>Top Ranked words</b> flying : 31 top : 33 airplanes : 344 motor : 314		<b>Word annotations :</b> <b>Word Rank</b> betty : 71215 exit : 1534 <b>Top Ranked words</b> street : 5 flag : 354 sky : 104 fence : 112

Fig. 8: Qualitative unsuccessful results of generated word rankings. Word instances without any semantic correlation with the visual information in the scene tend to be low ranked.

the use of the rich visual information contained in scene images that could provide the missing context to improve text detection and recognition results.

We have shown how in large scale datasets consisting in images and associated textual information, like image captions and scene text transcriptions, the topic modeling statistical framework can be used to leverage the correlation between visual and textual information in order to predict the words that are more likely to appear in the image as scene text instances. Moreover, we have shown that is possible to train a deep CNN model to reproduce those topic model based word rankings but using only an image as input.

Our experiments demonstrate that the quality of the automatically obtained custom lexicons is superior to a generic frequency based baseline, and thus can be used to improve scene text recognition methods. Future work will be devoted to integrate the proposed method in an end-to-end scene text reading system.

As a result of the work presented in this paper we have developed a cross API, which is made publicly available<sup>3</sup>, to get captions from MS-COCO and corresponding word annotations from COCO-Text.

## 6 Acknowledgments

This project was supported by the Spanish projects TIN2014-52072-P and RYC-2009-05031.

<sup>3</sup> [https://github.com/yash0307/MS-COCO\\_Text\\_CrossAPI](https://github.com/yash0307/MS-COCO_Text_CrossAPI)

## References

1. Karatzas, D., Shafait, F., Uchida, S., Iwamura, M., i Bigorda, L.G., Mestre, S.R., Mas, J., Mota, D.F., Almazan, J.A., de las Heras, L.P.: Icdar 2013 robust reading competition. In: 2013 12th International Conference on Document Analysis and Recognition, IEEE (2013) 1484–1493
2. Karatzas, D., Gomez-Bigorda, L., Nicolaou, A., Ghosh, S., Bagdanov, A., Iwamura, M., Matas, J., Neumann, L., Chandrasekhar, V.R., Lu, S., et al.: Icdar 2015 competition on robust reading. In: Document Analysis and Recognition (ICDAR), 2015 13th International Conference on, IEEE (2015) 1156–1160
3. Li, H., Shen, C.: Reading car license plates using deep convolutional neural networks and lstms. arXiv preprint arXiv:1601.05610 (2016)
4. Jaderberg, M., Simonyan, K., Vedaldi, A., Zisserman, A.: Reading text in the wild with convolutional neural networks. International Journal of Computer Vision **116**(1) (2016) 1–20
5. Gomez-Bigorda, L., Karatzas, D.: Textproposals: a text-specific selective search algorithm for word spotting in the wild. arXiv preprint arXiv:1604.02619 (2016)
6. Wang, T., Wu, D.J., Coates, A., Ng, A.Y.: End-to-end text recognition with convolutional neural networks. In: Pattern Recognition (ICPR), 2012 21st International Conference on, IEEE (2012) 3304–3308
7. Wang, K., Belongie, S.: Word spotting in the wild. In: European Conference on Computer Vision, Springer (2010) 591–604
8. Frinken, V., Karatzas, D., Fischer, A.: A cache language model for whole document handwriting recognition. In: Document Analysis Systems (DAS), 2014 11th IAPR International Workshop on, IEEE (2014) 166–170
9. Xiu, P., Baird, H.S.: Towards whole-book recognition. In: Document Analysis Systems, 2008. DAS'08. The Eighth IAPR International Workshop on, IEEE (2008) 629–636
10. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. Journal of machine Learning research **3**(Jan) (2003) 993–1022
11. Ye, Q., Doermann, D.: Text detection and recognition in imagery: A survey. IEEE transactions on pattern analysis and machine intelligence **37**(7) (2015) 1480–1500
12. Neumann, L., Matas, J.: On combining multiple segmentations in scene text recognition. In: 2013 12th International Conference on Document Analysis and Recognition, IEEE (2013) 523–527
13. Jaderberg, M., Vedaldi, A., Zisserman, A.: Deep features for text spotting. In: European conference on computer vision, Springer (2014) 512–528
14. Bissacco, A., Cummins, M., Netzer, Y., Neven, H.: Photoocr: Reading text in uncontrolled conditions. In: Proceedings of the IEEE International Conference on Computer Vision. (2013) 785–792
15. Mishra, A., Alahari, K., Jawahar, C.: Top-down and bottom-up cues for scene text recognition. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, IEEE (2012) 2687–2694
16. Milyaev, S., Barinova, O., Novikova, T., Kohli, P., Lempitsky, V.: Image binarization for end-to-end text understanding in natural images. In: 2013 12th International Conference on Document Analysis and Recognition, IEEE (2013) 128–132
17. Gómez, L., Karatzas, D.: Scene text recognition: No country for old men? In: Asian Conference on Computer Vision, Springer (2014) 157–168
18. Roy, U., Mishra, A., Alahari, K., Jawahar, C.: Scene text recognition and retrieval for large lexicons. In: Asian Conference on Computer Vision, Springer (2014) 494–508

19. Movshovitz-Attias, Y., Yu, Q., Stumpe, M.C., Shet, V., Arnoud, S., Yatziv, L.: Ontological supervision for fine grained classification of street view storefronts. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE (2015) 1693–1702
20. Feng, Y., Lapata, M.: Topic models for image annotation and text illustration. In: Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Association for Computational Linguistics (2010) 831–839
21. Papadimitriou, C.H., Tamaki, H., Raghavan, P., Vempala, S.: Latent semantic indexing: A probabilistic analysis. In: Proceedings of the seventeenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems, ACM (1998) 159–168
22. Fei-Fei, L., Perona, P.: A bayesian hierarchical model for learning natural scene categories. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). Volume 2., IEEE (2005) 524–531
23. Zhang, D., Luo, T., Wang, D., Liu, R.: Learning from lda using deep neural networks. arXiv preprint arXiv:1508.01011 (2015)
24. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531 (2015)
25. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. arXiv preprint arXiv:1512.00567 (2015)
26. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European Conference on Computer Vision, Springer (2014) 740–755
27. Veit, A., Matera, T., Neumann, L., Matas, J., Belongie, S.: Coco-text: Dataset and benchmark for text detection and recognition in natural images. arXiv preprint arXiv:1601.07140 (2016)
28. Rehurek, R., Sojka, P.: Software framework for topic modelling with large corpora. In: In Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks. (2010)
29. Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., et al.: Tensorflow: Large-scale machine learning on heterogeneous distributed systems. arXiv preprint arXiv:1603.04467 (2016)