Improving text recognition using optical and language model writer adaptation

Yann Soullard, Wassim Swaileh, Pierrick Tranouez, Thierry Paquet
LITIS lab
Normandie University, Universite de Rouen Normandie
76800 Saint Etienne du Rouvray, France
Email: {firstname.lastname}@univ-rouen.fr

Clément Chatelain
LITIS lab
INSA Rouen Normandie
76800 Saint Etienne du Rouvray, France
Email: clement.chatelain@insa-rouen.fr

***Abstract*—State-of-the-art methods for handwriting text recognition are based on deep learning approaches and language modeling that require large data sets during training. In practice, there are some applications where the system processes mono-writer documents, and would thus benefit from being trained on examples from that writer. However, this is not common to have numerous examples coming from just one writer. In this paper, we propose an approach to adapt both the optical model and the language model to a particular writer, from a generic system trained on large data sets with a variety of examples. We show the benefits of the optical and language model writer adaptation. Our approach reaches competitive results on the READ 2018 data set, which is dedicated to model adaptation to particular writers.**

***Keywords*-Handwriting recognition; writer adaptation; deep neural network; optical model; language model**

I. Introduction

Deep learning approaches combined with language modeling are now state-of-the art approaches for handwriting recognition tasks. However, they require large amounts of labeled training data to achieve optimal performance. In speech and handwriting recognition, many personalized applications have to deal with one specific handwriting document or audio recording coming from one writer or speaker. Besides, in the field of digital humanities, there is a need for digitization and transcriptions of old manuscripts, many of which being from one writer only.

Designing a handwriting recognition system trained to recognize the writing and language style of some specific writers is attractive, but incompatible with the large amount of examples required to train deep architectures and language models. Another difficulty is that speech and handwriting recognition systems may also suffer from a shift in the distribution of the target language between the training data set and the test data set.

As a consequence, there is a growing interest in designing learning algorithms that can be trained on small labeled data sets from the targeted domain, while keeping a good generalization ability. Domain adaptation [1] and transfer learning [2], [3] are machine learning approaches that specifically address the problem of data distribution shift between data sets that have some similar properties. Transfer learning is now a widespread approach that has shown to be very efficient on various recognition tasks such as handwriting recognition, signature identification or medical imaging [4], [5]. In addition, the language itself, but also lexicons as well as word frequencies may not match the statistical distributions learned during language model training. Statistical language model adaptation algorithms [6] have also been designed in this respect.

This paper presents an approach to adapt a generic text recognition system to a particular writer. In this respect, both the optical model and the language model of the generic system are considered during the adaptation phase. The generic system components (optical and language models) are first trained on a large data set (that does not contain any example of the writer). The optical model is based on a reference architecture that can benefit from transfer learning and data augmentation. The language model benefits from sub-lexical modeling [7] and model interpolation to circumvent the lack of training examples from the writer.

Our approach is evaluated on the 2018 READ competition [8] data sets, dedicated to writer adaptation analysis. We show that our method reaches competitive performance on the adaptation task and outperforms the systems which have been submitted to the competition. To the best of our knowledge, it is the first time that both an optical model and a language model adaptation is evaluated simultaneously for an adaptation task.

The article is organized as follows. In section II, we discuss related works on handwriting recognition system adaption, both for optical model and language model. Section III presents the proposed strategy for writer adaptation. Finally, experiments on the 2018 READ competition are described in section IV.

II. Related Works

Speech and handwriting recognition both suffer from the large variability between speakers or writers, and for both the acoustic or optical model and the language model. This variability is known to limit the good recognition performance of a generic multi-writer (speaker) system. Adaptation of a generic recognition system to a specific data set should therefore consider the adaptation of both the optical model and the language model. This is the proposition of this

paper. Note that language model adaptation has been studied extensively in speech recognition [9], but only few works have focused on handwriting recognition until now.

A. Optical model adaptation

The adaptation strategy is either a data normalization process [10] or a model adaptation process that modifies the parameters of a general-purpose recognizer to the specific speech or handwriting. In this respect, MLLR (Maximum Likelihood Linear Regression) [11] and MAP (Maximum A Posteriori) [12] have been proposed within the well established framework of Hidden Markov Models (HMM) to adapt the parameters of the acoustic/optical models, or their structure [13]. It is to be noticed also that most of the state-of-the-art industrial OCR rely on some adaptation principles [14] by introducing heuristic rules or statistics gathered at the document level.

Recently, as part of the ICFHR 2018 READ competition [8], most of the participants proposed an optical model composed of a Convolutional Neural Network (CNN) and Bidirectional Long Short Term Memory (BLSTM) layers as in [15] and [16]. One of them was using Multi-dimensional LSTM (MDLSTM) [17], which has provided good performance when trained on a generic large data set, but has shown difficulty in carrying the writer adaptation process with few samples. After the competition, [18] proposed a fully convolutional network architecture trained with the CTC loss function for text recognition. This system achieves the best results on the READ 2018 data set, significantly outperforming the other systems. Following the guidelines of the READ 2018 competition, every systems are trained on a generic data set, and then fine-tuned on specific documents using transfer learning and data augmentation strategies.

B. Language model adaptation

Language model (LM) adaptation has been studied extensively in speech recognition. In practice, there is rarely enough data to get a reliable language model learned on samples of one single writer's only [6]. Thus, language model adaptation consists in deriving a specific language model using a generic language model and a small training data set (text samples of one writer). The generic LM is previously trained on a much larger and general data set. Language model adaptation is commonly performed by interpolating the two models (specific and generic LM) [19], typically using a linear interpolation. In such approaches, a back-off to the generic LM can also be used as a way to fill up missing statistics of the specific LM. A back-off threshold has to be determined in this case.

In speech recognition, interpolations between generic LM and task-specific uni-grams have been proposed in [20]. In [21], the authors proposed to interpolate the n-gram probabilities of a generic and specific LM. In [6], weights interpolating multiple predefined LMs are trained. Regarding handwriting recognition, Xiu and Baird [22] adapted a word lexicon from OCR results. Lee and Smith [23] modified word uni-gram probabilities in caches. Besides, Wang *et al.* [19] presented three methods for language model adaptation in order to match a writer corpus with a predefined set of generic LM.

Finally, some of the participants of the READ 2018 competition proposed to use language models, such as n-grams of words, sub-lexical units or characters. In two systems, language models are the result of the interpolation of a document (writer) specific language model with a generic one. For both systems, balance between the two LM is set beforehand.

III. Proposed approach

We present in this section our strategy to adapt a generic handwriting recognition system to a specific writer. In the following we will call *generic data set* a large data set composed of multiple documents from different writers and languages, and *specific data set* a smaller data set coming from a document (or a subset of documents) written by a single writer. Similarly, we will call *generic model* and *specific model* an optical model or a language model trained on the generic data set or the specific data set, respectively.

A. System overview

The system architecture is illustrated in Figure 1. We first train a generic handwriting recognition system composed of an optical model ($P_G(O|W)$) and a language model ($P_G(W)$). This system is trained on the generic data set, in order to model the variability between different writing styles, languages and document images background. Note that we showed in [24] that a language model can account for numerous languages without affecting the performance, thus providing a multilingual LM.

Both generic models (optical and language) serve as initial models of the adaptation process intended to design specific optical ($P_S(O|W)$) and language models ($P_S(W)$). At decoding time, the outputs of the optical model, either generic or specific, are analyzed by the appropriate language model by exploring a search graph thanks to a Weighted Finite State Transducer (WFST) and using the Viterbi algorithm. The method selects the sentence $\hat{W}$ maximizing the *a posteriori* probability $P(W|O)$ among all possible sentences W by applying the Bayes formula:

$$\hat{W} = \arg\max_{w} P(W|O) = \arg\max_{w} P(O|W)P(W)^{\alpha}\beta^{|W|} \quad (1)$$

where O is the observation sequence coming from the input image, $P(O|W)$ is the probability of the observation sequence given the sentence W computed thanks to the optical character model and $P(W)$ is the prior probability of the sentence computed using the language model. The two hyper-parameters, α and β, are the language model

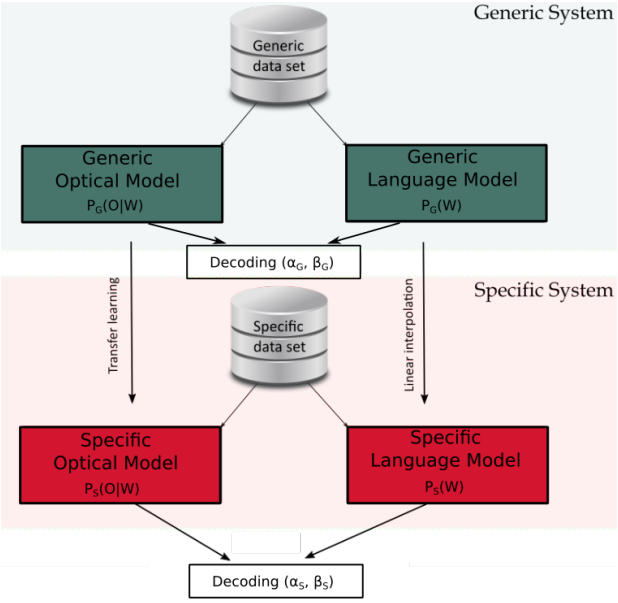

Figure 1. System architecture for writer adaptation.

scaling parameter and the word insertion penalty parameter, respectively.

B. Generic system

Both components of the generic system, the optical model (OCR) and the language model (LM), are inspired by recent advances in the field.

1) Optical model: The optical model architecture is similar to state-of-the-art architectures that combine multiple convolution layers with recurrent layers [25]. Convolutional layers are used to extract local features from the input images, while recurrent layers captures long term sequential dependencies. Input text line images are processed using a sliding window that feed convolutional layers. Features from every windows are concatenated to build a sequence of feature vectors which is the input of the recurrent layers. The whole optical model is trained using the well-known Connectionist Temporal Classification algorithm [26] which aligns the label sequences on their corresponding feature sequences.

2) Data augmentation: Data augmentation has shown to be an important trick to increase the variability of the training data set, in order to improve the system generalization. From the raw text line images, we create new images by applying both slanting and scaling transformations. For slanting, we apply a rotation of at most ± 1.5 radian on the input image and pad it with the background pixel values to get a rectangular image. Regarding scaling, we pad the image with the background pixels on the left, right, top and bottom according to randomly chosen horizontal and vertical scaling factors.

3) Language model: Language models are based on sub-lexical units called multigrams. Multigrams are sequences of characters of variable length that are representative of the language. As shown in [24], multigrams bring many advantages compared to traditional word language models: 1) lexicon size is highly reduced; 2) multigrams are robust units to cope with Out Of Vocabulary words (OOV); 3) multigrams can be learned on any training corpus, without the need for any linguistic expertise. To conclude, language models based on multigrams have shown to be a good trade-off between word language models and character language models [7], [24]. We train a Hidden Semi-Markov Model [27], [28] for estimating the set of multigrams that cover the training corpus. Then, we train a n-gram language model of multigrams on the training corpus, using a standard back-off model with Kneser-Ney smoothing.

We estimate one Language Model per language in the training corpus (denoted *mono-LM*) as well as one generic Language Model estimated on the overall training corpus (denoted *multi-LM*). These models will serve for language identification as well as the initialization of the writer adaptation process.

C. Writer adaptation

We now detail the writer and language adaptation processes of the generic system.

1) Optical model adaptation: Writer adaptation of the optical model relies on transfer learning [3]: the generic optical model is used to initialize the specific optical model. Then, the specific optical model is trained on the specific writer data set only but using data augmentation (see section III-B2). Moreover, as we are facing very small specific training data sets, no additional data can be kept aside to serve for validating the adaptation process. Therefore, the optical model is trained on the whole specific data set using a cross-validation procedure to select the number of training epochs.

Writing size has also proven to be a significant parameter to consider. In this respect, we introduced stride adaptation of the sliding window. This is possible by estimating the frames per label ratio on each specific training set, and then adapting the stride accordingly.

2) Language model adaptation: Designing a language model on a small specific data set is not straightforward. Due to the lack of specific texts, LM estimations are not representative of a specifc writer language distribution. In case of writer adaptation, dealing with sub-lexical units (e.g. multigrams) is a strong advantage compared to the traditional word-based LM: - sub-lexical units appear more frequently in the training corpus than words - they are less diverse than words, and they have a better coverage of the language. As a consequence, a multigram based LM has a better modeling capacity of the writer's language than a traditional word language model. Nevertheless, a language

model estimated on a specific data set (i.e. the writer's corpus) may remain of limited capacity due to the small size of the corpus. To avoid this limitation, we define a specific language model as a linear interpolation of a writer-based specific LM denoted LM_w and a generic LM denoted LM_g:

$$LM = \lambda LM_w + (1 - \lambda) LM_g \quad (2)$$

where λ balances between the two language models. The generic language model LM_g is the generic mono-LM that can be associated to the corpus by choosing the generic mono-LM with the smallest perplexity, subject to a threshold. If no language is selected, the generic multi-LM is selected as LM_g.

In contrast to recent works that use a fixed value of λ [8], here λ is selected using cross validation on the specific training corpus. This allows to get a suitable value of λ using writer language examples. Besides, the two hyper-parameters, α and β from equation 1, are optimized on a generic validation data set, as the small amount of labeled examples is used for training.

3) Decoding: We also apply data augmentation on test data, at recognition time. This strategy produces multiple predictions per observation in test. We combine the multiple predictions using a ROVER algorithm [29], that has shown to be a robust strategy [30], [31]. The ROVER algorithm is made of two parts: an alignment module and a voting module. The alignment module provides a label transition network of minimal cost which is built using edit distance and using an iterative process with pairwise comparisons. The voting module consists in extracting the most probable label sequence from the transition network.

IV. Experimentation

We evaluate our approach on the READ data set that was proposed as part of the ICFHR 2018 Automated Handwritten Text Recognition competition [8].

A. Data and protocol

The ICFHR 2018 Automated Handwritten Text Recognition competition was aiming to evaluate writer adaptation performance of generic systems. The data set consists of a generic data set, composed of 17 heterogeneous documents (roughly 12,000 text lines from various time periods and languages), and 5 specific data sets written by only one writer. The test set contains the 5 specific documents, each document containing 15 pages of a writer that was not seen during training.

Model adaptation is encouraged in the competition by providing multiple adaptation data sets for each test writer. For each specific test set, 4 transcriptions have to be submitted using respectively 0, 1, 4 or 16 specific pages for adaptation. The 0 page case refers to the generic system (without model adaptation). Hence, 20 transcriptions are submitted: 4 adaptation scenarios for the 5 specific documents. The Character Error Rate (CER) was used to evaluate the systems on each writer data set under each adaptation conditions, and finally the average CER was computed on the whole test sets.

B. Experimental setting

During training data augmentation was used to increase the generic data set by 10, while we decided to increase the specific data by 100 during adaptation. Indeed, there is a need to have enough adaptation data to carry out the adaptation process. Data augmentation is also introduced on the test sets, producing 19 augmented copies of each example. The recognition system provides 20 predictions for each example and its 19 augmented versions. The multiple predictions are combined using ROVER at character level, so as to provide a unique prediction for each text line.

The optical character recognition model is composed of 8 convolutional layers (similarly to VGG16 [32]) with max-pooling and dropout after two convolutions, followed by two BLSTM layers [33] and a dense layer with softmax activation in order to get a probability per label at each frame. A sliding window of 32 pixels width and 64 pixels height is applied in the writing direction. Stride adaptation leads to a stride of 4 for two writers (*Konzil C* and *Schiller*) and a stride of 2 for the 3 others, as the number of frames per label is under 20. The optical model is implemented in Keras [34] and trained using the Adam optimizer [35]. We performed a 6-fold cross validation on the specific adaptation set to prevent overfitting the optical models.

We choose a 9-gram language model of 2-multigrams. There is one LM per specific document and 4 generic mono-LM (one per language in the generic data set i.e. German, English, Danish and Swedish) in addition to one generic multi-LM. A threshold equal to 150 is used to select the mono-LM using the perplexity. Language models are estimated using the MIT language modeling toolkit [36] and the modified Kneser-Ney smoothing method [37] to estimate the back-off coefficients. Viterbi two-pass decoding is applied. The λ parameter (eq. 2) is the one providing the lowest average perplexity on a 6-fold cross validation on the specific set. Then, language model scale and word insertion penalty (see eq. 1) are optimized on a generic validation set.

C. Results

We compare our results with those of the other participants of the 2018 ICFHR competition: OSU, ParisTech, PRHLT, RPPDI and LITIS (our previous work). Every participant have used optical models based on convolutional and recurrent layers, except RPPDI that uses a MDLSTM network. n-gram LM were proposed by every participants, except for OSU. LM are based on interpolation scheme with a fixed value for λ, whatever the specific document. Data augmentation was performed both in training and testing by OSU and ParisTech, where the most frequent sequence of

Table I
AVERAGE CHARACTER ERROR RATE (CER) OF THE BEST SUBMITTED SYSTEMS COMPARED TO OUR PROPOSAL (THE LAST LINE). WE ALSO EVALUATE THE IMPACT OF EACH ADAPTATION (OPTICAL MODEL AND LANGUAGE MODEL G OR S FOR GENERIC AND SPECIALIZED). THIS WORK RELATES TO BOTH OPTICAL AND LANGUAGE MODEL SPECIALIZATION.

	CER per additional specific training pages					CER per specific test document					
	0	1	4	16	Imp	*Konzil C*	*Schiller*	*Ricordi*	*Patzig*	*Schwerin*	total CER
OSU	31.399	17.734	13.267	9.024	28.7	9.394	21.097	23.266	23.171	12.985	17.856
ParisTech	32.252	19.798	16.979	14.721	45.6	10.494	19.047	35.596	23.831	17.020	20.938
LITIS	35.294	22.508	16.887	11.345	32.1	9.139	25.692	30.501	25.184	18.041	21.508
PRHLT	32.793	22.157	17.895	13.329	40.6	8.651	18.393	35.069	26.257	18.653	21.541
RPPDI	30.805	28.404	27.246	22.846	74.1	11.901	21.880	37.292	32.752	28.553	27.325
FCI	**25.347**	**12.628**	**8.279**	5.825	22.9	6.490	**13.766**	**17.330**	**14.845**	12.329	**13.020**
om G + lm S	26.556	25.532	25.129	24.699	93.01	9.046	21.314	32.539	32.263	26.993	25.479
om S	28.484	16.329	10.495	6.251	**21.9**	7.331	16.913	21.940	19.686	**11.519**	15.390
om S + lm G	26.556	16.122	10.807	6.651	25.1	6.036	14.787	23.121	18.178	12.909	15.034
this work	26.556	15.472	9.999	**5.819**	**21.9**	**5.940**	14.811	21.616	18.081	**11.793**	14.458

labels per data was selected to be the output of the system. Finally, the system proposed by the Faculty of Computers and Information of Asyut (FCI) has been submitted after the deadline. Their approach is based on a deep fully convolutional network and many data augmentation strategies (projective transforms, elastic distortions, sign flipping). This approach significantly outperforms the systems submitted during the competition.

Table I shows the results of our method compared to the other methods that have been submitted. We achieve the best improvement from 0 to 16 specific training pages with nearly 78% decrease of the CER. Our method also outperforms the other ones submitted to the competition on every metric (reported in [8]). While, the method proposed by [18] seems relevant when it is trained without many specific examples, our approach reaches state of the art performance on writer adaptation using 16 specific training pages.

Table I also highlights the improvement of each model adaptation (optical and LM). One notes that the system is very poor without optical model specialization as the specialized LM provides a small improvement only. In case of a specialized optical model, the use of a generic LM may benefit the optical model, especially when it is trained on a small amount of specific pages. However, the shift between the generic and writer language distinctiveness seems to be too important and the use of a language model may have a negative impact. This is illustrated on the Ricordi data set, as there is no Italian text in the generic set and on the Schwerin data set, which is the German specific document for which the German generic mono-LM provides the higher perplexity value. Finally, adding a specialized LM on the output of a specialized optical model increases the recognition rate nearly 1 point of percentage in CER.

Besides, as stated by [8], it is reasonable that a human corrects a transcription with a CER below 10%. In this regard, Table II shows the CER per specific document when using 16 specific pages in training. Our method is the only one that achieves a CER under 10% on each specific document, with a CER under 3% for two of them. This shows the interest of our method to real-life use cases, when a perfect transcription is required and that automatic transcription will be corrected by humans.

Table II
CER PER DOCUMENT WITH 16 SPECIFIC PAGES IN TRAINING.

	CER per specific test document					
	Konzil C	*Schiller*	*Ricordi*	*Patzig*	*Schwerin*	< 10
OSU	**3.79**	12.45	15.04	12.54	**3.50**	2
ParisTech	**8.02**	14.58	30.19	15.51	**9.18**	2
LITIS	**4.81**	19.57	16.37	12.83	**6.61**	2
PRHLT	**4.98**	12.55	28.52	16.35	**7.12**	2
RPPDI	**9.18**	16.29	30.49	28.30	24.90	1
FCI	**2.83**	**8.17**	11.44	**6.73**	**2.28**	4
this work	**2.73**	8.41	**9.72**	**7.19**	**2.74**	**5**

V. CONCLUSION

In this paper, we proposed to carry out writer adaptation of a whole handwriting recognition system by adapting both the optical model and the language model. The proposed approach is evaluated on the ICFHR 2018 competition using the READ data set and reaches state-of-the art performance during writer adaptation. We obtain the best performance both with 16 specific training pages to adapt the generic system and the best improvement from the generic system to the specific one. Our approach is the first one that reaches a character error rates below 10% on each specific data set, which was established to be the value below which human correction can be conducted with acceptable efforts.

REFERENCES

[1] L. Y. Pratt, "Discriminability-based transfer between neural networks," in *NIPS*, 1993, pp. 204–211.

[2] C. B. Do and A. Y. Ng, "Transfer learning for text classification," in *NIPS*, 2006, pp. 299–306.

[3] S. J. Pan, Q. Yang *et al.*, "A survey on transfer learning," *IEEE Transactions on knowledge and data engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[4] D. C. Cireşan, U. Meier, and J. Schmidhuber, "Transfer learning for latin and chinese characters with deep neural networks," in *IJCNN*, 2012, pp. 1–6.

[5] S. Belharbi, C. Chatelain, R. Hrault, S. Adam, S. Thureau, M. Chastan, and R. Modzelewski, "Spotting L3 slice in CT scans using deep convolutional network and transfer learning," *Computers in Biology and Medicine*, 2017.

[6] J. R. Bellegarda, "Statistical language model adaptation: review and perspectives," *Speech communication*, vol. 42, no. 1, pp. 93–108, 2004.

[7] W. Swaileh, T. Paquet, Y. Soullard, and P. Tranouez, "Handwriting recognition with multigrams," in *ICDAR*, vol. 1, 2017, pp. 137–142.

[8] T. Strauss, G. Leifert, R. Labahn, H. T., and G. Muhlberger, "Icfhr2018 competition on automated text recognition on a read dataset," in *ICFHR*, 2018, pp. 477–482.

[9] J. Kolář, Y. Liu, and E. Shriberg, "Speaker adaptation of language models for automatic dialog act segmentation of meetings," in *Interspeech*, vol. 1, 2007.

[10] J. S. Bridle and S. J. Cox, "Recnorm: Simultaneous normalisation and classification applied to speech recognition," in *NIPS*, 1991, pp. 234–240.

[11] C. J. Leggetter and P. C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden markov models," *Computer speech & language*, vol. 9, no. 2, pp. 171–185, 1995.

[12] J. luc Gauvain and C. hui Lee, "Maximum a posteriori estimation for multivariate gaussian mixture observations of markov chains," *IEEE Transactions on Speech and Audio Processing*, vol. 2, pp. 291–298, 1994.

[13] K. Ait-Mohand, T. Paquet, and N. Ragot, "Combining structure and parameter adaptation of hmms for printed text recognition," *IEEE PAMI*, vol. 36, no. 9, pp. 1716–1732, 2014.

[14] I. Marosi, "Industrial ocr approaches: architecture, algorithms, and adaptation techniques," in *Proc. SPIE. 6500, Document Recognition and Retrieval XIV*. SPIE, 2007, pp. –.

[15] B. Shi, X. Bai, and C. Yao, "An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition," *IEEE PAMI*, vol. 39, no. 11, pp. 2298–2304, 2017.

[16] E. Chammas, C. Mokbel, and L. Likforman-Sulem, "Handwriting recognition of historical documents with few labeled data," in *DAS*. IEEE, 2018, pp. 43–48.

[17] A. Graves and J. Schmidhuber, "Offline handwriting recognition with multidimensional recurrent neural networks," in *NIPS*, 2009, pp. 545–552.

[18] M. Yousef, K. F. Hussain, and U. S. Mohammed, "Accurate, data-efficient, unconstrained text recognition with convolutional neural networks," *arXiv:1812.11894*, 2018.

[19] Q.-F. Wang, F. Yin, and C.-L. Liu, "Unsupervised language model adaptation for handwritten chinese text recognition," *Pattern Recognition*, vol. 47, no. 3, pp. 1202–1216, 2014.

[20] Y. Akita and T. Kawahara, "Language model adaptation based on plsa of topics and speakers," in *Eighth International Conference on Spoken Language Processing*, 2004.

[21] G. Tur and A. Stolcke, "Unsupervised language model adaptation for meeting recognition," in *ICASSP*, vol. 4, 2007, pp. IV–173.

[22] P. Xiu and H. S. Baird, "Whole-book recognition," *IEEE PAMI*, vol. 34, no. 12, pp. 2467–2480, 2012.

[23] D.-S. Lee and R. Smith, "Improving book ocr by adaptive language and image models," in *DAS*, 2012, pp. 115–119.

[24] W. Swaileh, Y. Soullard, and T. Paquet, "A unified multilingual handwriting recognition system using multigrams sub-lexical units," *Pattern Recognition Letters*, 2018.

[25] D. Suryani, P. Doetsch, and H. Ney, "On the benefits of convolutional neural network combinations in offline handwriting recognition," in *ICFHR*, 2016, pp. 193–198.

[26] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *ICML*. ACM, 2006, pp. 369–376.

[27] K. P. Murphy, "Hidden semi-markov models (hsmms)," *unpublished notes*, vol. 2, 2002.

[28] S.-Z. Yu, "Hidden semi-markov models," *Artificial Intelligence*, vol. 174, no. 2, pp. 215–243, 2010.

[29] J. G. Fiscus, "A post-processing system to yield reduced word error rates: Recognizer output voting error reduction (rover)," in *Automatic Speech Recognition and Understanding*, 1997.

[30] T. Bluche, J. Louradour, M. Knibbe, B. Moysset, M. F. Benzeghiba, and C. Kermorvant, "The a2ia arabic handwritten text recognition system at the open hart2013 evaluation," in *DAS*. IEEE, 2014, pp. 161–165.

[31] B. Stuner, C. Chatelain, and T. Paquet, "Lv-rover: lexicon verified recognizer output voting error reduction," *arXiv preprint arXiv:1707.07432*, 2017.

[32] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[33] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[34] Y. Soullard, C. Ruffino, and T. Paquet, "Ctcmodel: a keras model for connectionist temporal classification," *arXiv preprint arXiv:1901.07957*, 2019.

[35] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[36] B.-J. P. Hsu, "Language modeling for limited-data domains," Ph.D. dissertation, MIT, 2009.

[37] R. Kneser and H. Ney, "Improved backing-off for m-gram language modeling," in *ICASSP*, vol. 1, 1995, pp. 181–184.