The UPV Handwriting Recognition and Translation System for OpenHaRT 2013

Ihab Khoury, Adrià Giménez, Jesús Andrés-Ferrer, Alfons Juan and Joan Andreu Sánchez
DSIC/ITI, Universitat Politècnica de València, Spain
{ialkhoury,agimenez,jandres,ajuan,jandreu}@dsic.upv.es

***Abstract*—The NIST Open Handwriting Recognition and Translation Evaluation 2013 (NIST OpenHaRT'13) is a performance evaluation assessing technologies that transcribe and translate text in document images. This evaluation is focused on recognizing Arabic text images and translating them into English. A Handwriting Recognition and Translation system typically consists of a combination of two systems: a Text Recognition system and a Machine Translation system. In this paper, we present the *UPV* participation in the NIST OpenHaRT 2013 evaluation. For the Text Recognition system we used the TL toolkit for training and recognition. For the Machine Translation system we used the Moses toolkit for training and decoding. Results in this evaluation are challenging and they significantly outperform our previous results in the OpenHaRT 2010 evaluation.**

Keywords—NIST OpenHaRT, Arabic HTR, Bernoulli HMM, Sliding Window, Repositioning

I. Introduction

To our knowledge, there are only a few systems that are able to automatically translate handwritten text images into another language, in particular, Arabic. Typically, the available systems are based on a concatenation of two systems: a Handwritten Text Recognition (HTR) system and a Machine Translation (MT) system. The NIST OpenHaRT'13 evaluation [1] is aimed at assessing systems that recognize Arabic handwritten text images, and then translate the recognized handwritten images into English. In this paper, we describe the UPV systems presented in the evaluation campaign. Roughly speaking, in the case of handwritten recognition of text images, our work has focused on the use of the *embedded Bernoulli (mixture) HMMs (BHMMs),* that is, embedded HMMs in which the emission probabilities are modeled with Bernoulli mixtures [2]. In the case of Arabic text translation, our work has focused on one of the state-of-the-art phrase-based log-linear translation models, Moses [3]. In what follows, we briefly review the UPV transcription (Sec. II), and translation systems (Sec. III). After that, we outline the submitted systems and their results in Sections IV and V, as well as the employed tools in Sec. VI. Concluding remarks are given in Sec. VII.

II. Transcription System

The UPV system is based on windowed BHMMs (Bernoulli HMMs) [4], [2], [5]. Each transcription hypothesis is built from an HMM in which emission probabilities are modelled as Bernoulli mixture distributions. To keep the number of independent parameters low, the BHMM at sentence level (transcription hypothesis) is built from BHMMs at character level which depend on their surrounding characters, the so-called tri-character modelling approach. Given a text image of an unknown word, each windowed BHMM computes the probability of the given image to be a handwritten version of its corresponding word. To compute these probabilities, text images are first transformed into a sequence of binary feature vectors by applying a sliding window at each horizontal position. The width of the sliding window is known to have a strong effect on the system ability to capture local image distortions, and thus this parameter has to be tuned. Moreover, we have recently observed that local image distortions, and vertical distortions in particular, might not be properly modeled when the sliding window is applied at a constant vertical position of the image. To overcome this limitation, we applied *repositioning* on the sliding window before its actual application. That is, the sliding window was repositioned so as to align its center with its mass center. In this work, we applied only a vertical repositioning due to its better performance over another two methods (horizontal and in both directions) discussed in [6], [7], [8]. In Figure. 1, the standard method (no repositioning) is compared with vertical repositioning. More details on this idea of repositioning are discussed in [8].

The UPV system was trained from input images scaled in height to 30 pixels (while keeping the aspect ratio), and then binarized with the Otsu algorithm [9]. A sliding window of width 9 using the vertical repositioning was applied, and thus the resulting input (binary) feature vectors for the BHMMs had 270 bits. Since in Arabic, the shape of a letter written at the beginning of the word is different from a letter written at the middle or at the end; all Arabic transcriptions were encoded by adding this shape information.

Finally, the number of states per character was adjusted to 6 states for all BHMMs. Similarly, the number of mixture components per state was empirically adjusted to 128. Parameter estimation and recognition were carried out using the EM algorithm. Also, we used a 5-gram language model at character level instead of the conventional class priors. The language model was smoothed by linear interpolated estimates with absolute modified Kneser-Ney discounting. In addition, the grammar scale factor was adjusted to 30.

III. Translation System

The UPV system for the translation task is based on a state-of-the-art log-linear translation system, specifically, using Moses toolkit [3]. Nowadays, SMT systems follow the Bayes

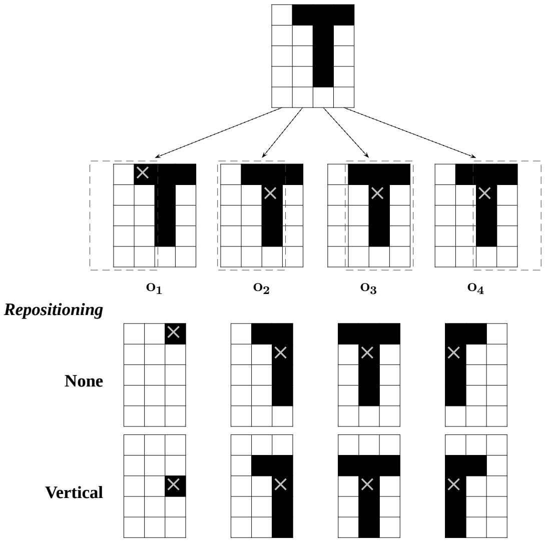

Fig. 1. Example of transformation of a 4×5 binary image (top) into a sequence of 4 15-dimensional binary feature vectors $O = (\mathbf{o}_1, \mathbf{o}_2, \mathbf{o}_3, \mathbf{o}_4)$ using a window of width 3. After window extraction (illustrated under the original image), the standard method (no repositioning) is compared with the vertical repositioning. Mass centers of extracted windows are also indicated.

decision rule approach [10], [11] in which the optimal target sentence y is found by maximizing the posterior probability,

$$y^* = \operatorname*{argmax}_y \; p(y \mid x), \tag{1}$$

where the posterior probability is modeled as a log-linear combination of feature functions [12] as follow

$$y^* = \operatorname*{argmax}_y \sum_{m=1}^{M} \lambda_m h_m(x, y), \tag{2}$$

with λ_m being the log-linear interpolation weight and $h_m(x, y)$ is a feature function, such as the logarithm of a language model, or the logarithm of a phrased-based model. Specifically, in our system, we used the standard Moses features: a phrased-based model that includes both direct and inverse phrase translation probabilities and both direct and inverse lexical weights, a language model, a distance-based reordering model, a word penalty, and a lexicalized reordering model. In the case of the language model, we used a *5-gram* model trained with SRILM [13]. This model was smoothed by linear interpolated estimates with absolute modified Kneser-Ney discounting.

Each source and target sentence was pre-processed. English text was tokenized with Moses tokenization tools [3], and Arabic text was tokenized using the *MADA+TOKAN* tool [14]. Additionally, long sentences (longer than 150 words) were then removed. Finally, standard Moses training was performed on the training data, which includes: alignment extraction, phrase extraction and MERT [3].

IV. Submissions

In this section, we describe the submissions made by the UPV team to the NIST OpenHaRT 2013 evaluation. Systems were submitted for three different tasks, the Document Image Recognition (DIR) task, the Document Text Translation (DTT) task, and the Document Image Translation (DIT) task. Systems were trained following two training conditions: a constrained condition that required participants to develop their systems using only the provided LDC data resources, and an unconstrained condition in which participants are free to use any additional publicly available non-LDC resources for the system development (For more information, please refer to [1]).

For the DIR task, the UPV submitted *two* systems (*DIR1*, the primary system, and *DIR2*, the contrastive system) that followed the constrained training condition. They used the BHMMs described in Sec. II. The only difference between the systems is that the DIR2 system was trained using the complete data set, whereas the DIR1 system was trained using less data. Statistics about the data used to train both systems (DIR1 and DIR2) are reported in Table I.

For the DTT, *two* primary systems were submitted. The first one followed the constrained training condition (DTT constrained), while the other one followed the unconstrained training condition (DTT unconstrained). Both systems were trained using the system described in Section. III. However, for the unconstrained task, we used some of the freely available data that was used in *IWSLT 2011* challenge: *MultiUN* [15] and *TED* [16]. Since *MultiUN* corpus is not aligned at sentence level, we used the Champollion [17] tool for aligning the sentences. Finally, we selected sentences for the training set according to the infrequent n-grams score [18], in order to gather a specific training set to translate our source test sentences. It is worth noting that the number of sentences used for training was $20K$ from MultiUN and $2K$ from TED. Further statistics about each corpus used to train our translation systems in both conditions (DTT constrained and DTT unconstrained) are shown in Table II. We used around $40K$ of data segments to train our system following the constrained condition. However, we used about $62K$ of data segments to train the DTT system following the unconstrained condition.

Given a handwritten image f, the DIT task, can be expressed as follows,

$$y^\star = \operatorname*{argmax}_{y \in Y} \; p(y|f) = \operatorname*{argmax}_{y \in Y} \sum_x p(x|f) \; p(y|x) \tag{3}$$

where x stands for a candidate recognized source (Arabic) text and y for a candidate translated sentence (in English) corresponding to the input image f.

Since the summation over all possible transcriptions in Eq. (3) cannot be computed in practice, for the Document Image Translation (DIT) task, we submitted three different systems. In all of them, the probability $p(x \mid f)$ in Eq. (3) was approximated by the primary DIR transcription system. Therefore, the key difference among systems lay in the translation subsystems.

In the primary DIT system (DIT1), Eq. (3) was approximated as follows,

$$\begin{aligned} y^* &\approx \operatorname*{argmax}_{y \in Y} \; [\max_x \{p(x|f) \; p(y|x)\}] \\ &\approx \operatorname*{argmax}_{y \in Y} \; [p(y| \max_x \{p(x|f)\})] \end{aligned} \tag{4}$$

and $p(y|x^\star)$ was approximated by the primary DTT translation system. In other words, the input image was recognized by the primary DIR transcription system, and the recognized text was fed into the primary DTT translation system.

The second DIT system (DIT2) followed a similar approach to that of the first DIT system, approximating Eq. (3) by Eq. (4). However, the translation probability was approximated by a translation system analogous to the primary DTT system but trained differently. In this case, the source part of each bilingual training pair was substituted by the transcription obtained by the primary DIR system. The new training data set produced in this way was used to train the translation system. This second translation system was expected to better handle the noisy output of the DIR system. Accordingly, this system showed a better performance than the standard (primary) system in the development set. However, in the test set it showed a worse performance. For further details, please refer to Table IV.

Finally, in the third system (DIT3), a different approximation of Eq. (3) was used

$$y^\star = \operatorname*{argmax}_{x\in\text{NBest}(f)} \left\{ \operatorname*{argmax}_{y\in\text{NBest}(f|x)} \{p(x|f)\ [p(y|x)]^\theta\} \right\} \qquad (5)$$

where we introduced a scaling factor θ, and the search space was approximated by N-best lists. Specifically, each input image was first recognized using the primary DIR system into 100-Best transcriptions, and then each transcription was translated using the primary DTT system into 100-Best translations. Finally, the optimal scaling factor θ was found using a grid search in a development set so as to maximize the BLEU.

In Tables I and II (last row), we report the data used to train each part of our Recognition and Translation System in the constrained condition. For the recognition part, we used about $779K$ of data lines for training, and for the translation part we used around $40K$ of data segments for training.

TABLE I. DATA (LINES) USED FOR TRAINING EACH SYSTEM AND ITS TRAINING CONDITIONS.

System/Condition	Constrained	Unconstrained
DIR1	$779,100$	-
DIR2	$789,874$	-
DIT (recognition part)	$779,100$	-

TABLE II. DATA (SEGMENTS) USED FOR TRAINING EACH SYSTEM AND ITS TRAINING CONDITIONS.

System/Condition	Constrained	Unconstrained	
Corpus	LDC	MultiUN	TED
DTT	$40,580$	$19,956$	$2,205$
DIT (translation part)	$40,580$	-	-

V. RESULTS

In this section, we summarize the results obtained in the OpenHaRT 2013 evaluation for all presented systems. For recognition systems, results are shown in terms of Word Error Rate (WER%), whereas for translation systems, results are shown in terms of BLEU score. In Table III, results for the two DIR systems (DIR1 and DIR2) are reported on the EVAL set [1] (Eval'13 column). Also, these systems, in particular DIR1, was compared with the OpenHaRT 2010 system (UPV PRHLT) for DIR and line segmentation condition. This comparison was performed by evaluating both systems on the DRYRUN set [1] (Eval'10 column). It is worth noting that the evaluation set in the OpenHaRT 2010 is the development set in the OpenHaRT 2013 (Eval'10 column). Having this in mind, we can easily compare our previous results obtained in OpenhaRT 2010 with results obtained in the DRYRUN set of the OpenHaRT 2013. On the other hand, Table IV reports results of the DTT system for both training conditions (constrained DTT and unconstrained DTT) together with the three DIT systems (DIT1, DIT2, and DIT3). These systems were evaluated on both sets EVAL and DRYRUN (Eval'10 and Eval'13 columns respectively). The evaluation on EVAL set was performed by NIST. However, the evaluation on DRYRUN set was performed by UPV. The UPV evaluation procedure might has slightly differed from the NIST procedure.

TABLE III. SUBMITTED SYSTEMS FOR DIR AND LINE SEGMENTATION CONDITION TOGETHER WITH THEIR WORD ERROR RATE (WER%)

System	Reference	WER [%]	
		Eval'10	Eval'13
DIR1	p-1_1_20130425	29.08	29.32
DIR2	c-1_2_20130425	-	**29.20**
UPV PRHLT	OpenHaRT'10	47.45	-

As shown in Table III, the DIR2 system slightly outperforms the DIR1 system. This conclusion was obviously expected for us since DIR2 system was trained with more data. Additionally, both DIR1 and DIR2 systems outperform our system (UPV PRHLT) from the OpenHaRT 2010 evaluation. This was also expected because in this evaluation we trained our models with more mixture components (128) per state, and also we used a bigger language model for recognition.

TABLE IV. SUBMITTED SYSTEMS FOR (DTT AND DIT) AND LINE SEGMENTATION CONDITION TOGETHER WITH THEIR BLEU SCORE

System	Reference	BLEU [%]	
		Eval'10	Eval'13
DTT Constrained	p-1_1_20130425	22.53	21.93
DTT Unconstrained	p-1_1_20130425	25.18	24.10
DIT1	p-1_1_20130425	16.51	16.95
DIT2	c-1_2_20130425	16.58	16.52
DIT3	c-1_3_20130425	18.13	**17.49**

As shown in Table IV, the usage of an additional small set of data (around $20K$) significantly improved the translation accuracy in the DTT system. More precisely, the Unconstrained DTT system significantly outperforms the Constrained DTT system. Here, we remind the reader that this additional data was selected according to the infrequent n-grams score [18], in order to gather a specific training set that relates to the source test sentences. In the same Table (IV), the DIT3 shows better performance than DIT1 and DIT2. Specifically, in the DIT3 system, the search space was approximated by means of 100-best list. This approach helped in finding better transcriptions and translations which resulted in improving the results.

VI. Tools and Means

In this section we describe the tools used in this work. For text pre-processing, we used the Moses tokenization tools [3] for English text tokenization. On the other hand, we used the MADA+TOKAN [14] toolkit for Arabic tokenization, diacritization, morphological disambiguation, POS tagging, stemming and lemmatization. In addition, we used the Champollion Toolkit (CTK) [17] to align the MultiUN [15] parallel corpus on sentence level.

For the handwritten text recognition system, we used the TLK [19] toolkit which among other features implements Bernoulli Hidden Markov models (BHMMs). This toolkit was developed by the UPV.

For the machine translation system, we used one of the state-of-the-art, phrase-based statistical machine translation systems, *Moses* [3]. To establish the word alignments of a parallel corpus, we used MGIZA++ [20].

For both handwritten text recognition and machine translation systems we used the SRI Language Modeling Toolkit (SRILM) [13] to generate the corresponding language models.

VII. Conclusion

In this paper, we described the UPV system submissions to the *NIST OpenHaRT'13* evaluation. Our submissions included systems for both transcription and translation. Specifically, two systems were submitted for the DIR task, one system for the DTT task, which followed both constrained and unconstrained training conditions, and three systems for the DIT task. Current results for the DIR task outperform previous results in OpenHaRT 2010 evaluation. Also, results for DTT and DIT tasks are very promising.

Acknowledgment

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement no 287755. Also supported by the Spanish Government (Plan E, iTrans2 TIN2009-14511 and AECID 2011/2012 grant).

References

[1] A. Tong, M. Przybocki, V. Maergner, and H. El Abed. Nist 2013 open handwriting recognition and translation (openhart'13) evaluation. *Proceedings of the NIST 2013 Open Handwriting and Recognition Workshop*, 2013. In press.

[2] Adrià Giménez, Ihab Khoury, and Alfons Juan. Windowed Bernoulli Mixture HMMs for Arabic Handwritten Word Recognition. In *Proceedings of the 12th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, Kolkata (India), November 2010.

[3] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, and R. Zens. Moses: Open source toolkit for statistical machine translation. In *Annual meeting-association for computational linguistics*, volume 45, page 2, 2007.

[4] A. Giménez and A. Juan. Embedded Bernoulli Mixture HMMs for Handwritten Word Recognition. In *Proc. of the 10th Int. Conf. on Document Analysis and Recognition (ICDAR 2009)*, pages 896–900, Barcelona (Spain), July 2009.

[5] Ihab Khoury, Adrià Giménez, and Alfons Juan. Arabic Handwritten Word Recognition Using Bernoulli Mixture HMMs. In *PICCIT '10*, Hebron (Palestine), March 2010.

[6] Ihab Khoury, Adri Gimnez-Pastor, Jess Andrs-Ferrer, and Alfons Juan-Cscar. Arabic Printed Word Recognition Using Windowed Bernoulli HMMs. In *Proc. of the 17th Int. Conf. on Image Analysis and Processing (ICIAP 2013)*, Naples (Italy), September 2013. Accepted.

[7] Ihab Khoury, Adri Gimnez, and Alfons Juan. Arabic handwriting recognition using bernoulli hmms. In Volker Mrgner and Haikal El Abed, editors, *Guide to OCR for Arabic Scripts*, pages 255–272. Springer London, 2012.

[8] Adrià Giménez, Ihab Khoury, Jesús Andrés-Ferrer, and Alfons Juan. Handwriting word recognition using windowed bernoulli hmms. *Pattern Recognition Letters*, 2013. In press.

[9] Nobuyuki Otsu. A Threshold Selection Method from Gray-Level Histograms. *IEEE Trans. on Systems, Man and Cybernetics*, 9:62–66, 1979.

[10] P. F. Brown, J. Cocke, S. A. D. Pietra, V. J. D. Pietra, F. Jelinek, J. D. Lafferty, R. L. Mercer, and P. S. Roossin. A statistical approach to machine translation. *Computational linguistics*, 16(2):7985, 1990.

[11] Jess Andrs Ferrer. *Statistical approaches for natural language modelling and monotone statistical machine translation*. PhD thesis, UNIVERSIDAD POLITCNICA DE VALENCIA, Valencia, Spain, February 2010.

[12] F. J. Och and H. Ney. The alignment template approach to statistical machine translation. *Computational Linguistics*, 30(4):417449, 2004.

[13] A. Stolcke. SRILM-an extensible language modeling toolkit. In *Proc. of the Inter. Conf. on spoken language processing*, volume 2, page 901904, 2002.

[14] Owen Rambow Nizar Habash and Ryan Roth. Mada+tokan: A toolkit for arabic tokenization, diacritization, morphological disambiguation, pos tagging, stemming and lemmatization. In Khalid Choukri and Bente Maegaard, editors, *Proc. of the 2nd Int. Conf. on Arabic Language Resources and Tools*, Cairo, Egypt, April 2009. The MEDAR Consortium.

[15] Andreas Eisele and Yu Chen. Multiun: A multilingual corpus from united nation documents. In Daniel Tapias, Mike Rosner, Stelios Piperidis, Jan Odjik, Joseph Mariani, Bente Maegaard, Khalid Choukri, and Nicoletta Calzolari (Conference Chair), editors, *Proc. of the Seventh conf. on Int. Language Resources and Evaluation*, pages 2868–2872. European Language Resources Association (ELRA), 5 2010.

[16] Ted corpus in the iwslt 2011 evaluation campaign, http://iwslt2011.org/doku.php?id=06_evaluation, 2011.

[17] X. Ma. Champollion: A robust parallel text sentence aligner. In *LREC 2006: Fifth International Conference on Language Resources and Evaluation*, page 489492, 2006.

[18] Guillem Gascó, Martha-Alicia Rocha, Germán Sanchis-Trilles, Jesús Andrés-Ferrer, and Francisco Casacuberta. Does more data always yield better translations? In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 152–161, Avignon, France, April 2012. Association for Computational Linguistics.

[19] The translectures-upv team. the translectures-upv toolkit (tlk). http://translectures.eu/tlk. http://www.translectures.eu/tlk/citing-tlk/, 2013.

[20] Qin Gao and Stephan Vogel. Parallel implementations of word alignment tool. In *In Proc. of the ACL 2008 Software Engineering, Testing, and Quality Assurance Workshop*, 2008.