International Journal of Document Analysis and Recognition manuscript No.
(will be inserted by the editor)

Confidence and Margin-Based MMI/MPE Discriminative Training for Offline Handwriting Recognition

Philippe Dreuw · Georg Heigold · Hermann Ney

Received: 16-02-2010 / Accepted: 04-02-2011

Abstract We present a novel confidence- and margin-based discriminative training approach for model adaptation of a hidden Markov model (HMM) based handwriting recognition system to handle different handwriting styles and their variations.

Most current approaches are maximum-likelihood (ML) trained HMM systems and try to adapt their models to different writing styles using writer adaptive training, unsupervised clustering, or additional writer specific data. Here, discriminative training based on the maximum mutual information (MMI) and minimum phone error (MPE) criteria are used to train writer independent handwriting models. For model adaptation during decoding, an unsupervised confidence-based discriminative training on a word and frame level within a two-pass decoding process is proposed.

The proposed methods are evaluated for closed-vocabulary isolated handwritten word recognition on the IFN/ENIT Arabic handwriting database, where the word-error-rate is decreased by 33% relative compared to a ML trained baseline system. On the large-vocabulary line recognition task of the IAM English handwriting database, the word-error-rate is decreased by 25% relative.

Keywords Handwriting Recognition · Arabic · Discriminative Training · Maximum Mutual Information · Minimum Phone Error · Margin · Confidences · Hidden Markov Models · Model Adaptation

RWTH Aachen University
Human Language Technology and Pattern Recognition
Ahornstr 55, D-52056 Aachen, Germany
Tel.: +49-241-80-21613
Fax: +49-241-80-22219
E-mail: dreuw@cs.rwth-aachen.de
Extended version of ICDAR 2009 work presented in [6]
The final publication is available at
http://www.springerlink.com

1 Introduction

Most state-of-the-art single-pass and multi-pass [4, 8, 11] HMM based handwriting recognition systems are trained using the maximum-likelihood (ML) criterion.

Typical training criteria for string recognition like for example minimum phone error (MPE) and maximum mutual information (MMI) in speech recognition are based on a (regularized) loss function. In contrast, large margin classifiers - the de-facto standard in machine learning - maximize the separation margin. An additional loss term penalizes misclassified samples.

The MMI training criterion has been used in [30] to improve the performance of an HMM based offline Thai handwriting recognition system for isolated characters. They propose a feature extraction based on a block-based PCA and composite image features, which are reported to better at discriminating Thai confusable characters. In [5], the authors apply the Minimum Classification Error (MCE) criterion to the problem of recognizing online unconstrained-style characters and words, and report large improvements on a writer-independent character recognition task when compared to a ML trained baseline system.

Similar to the system presented in [29], we apply the MMI criterion, but modified by a margin term. This margin term can be interpreted as an additional observation-dependent prior weakening the true prior [18], and is identical with the SVM optimization problem of log-linear models [16].

The most common way for unsupervised adaptation is the use of the automatic transcription of

a previous recognition pass without the application of confidence scores. Many publications in automatic speech recognition (ASR) have shown that the application of confidence scores for adaptation can improve recognition results. However, only small improvements are reported for confidence-based maximum-likelihood linear regression (MLLR) adaptation [12, 31, 33] or constrained-MLLR adaptation [1]. In this work, we present a novel unsupervised confidence-based discriminative model adaptation approach.

This paper briefly reviews how the MMI/MPE training criteria can be extended to incorporate the margin concept, and that such modified training criteria are smooth approximations to support vector machines with the respective loss function [16]. In addition to the margin concept, the MMI/MPE training criteria are extended by an additional confidence term [6] to allow for novel unsupervised model adaptation.

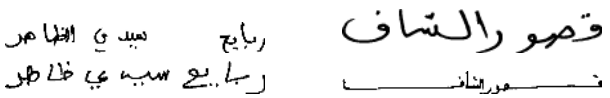
The focus of this work shall be on offline handwriting recognition of closed-vocabulary isolated words and large-vocabulary sentence recognition tasks in combination with m-gram language models. More explicitly, the novelties of our investigation are as follows:

1. Direct evaluation of the utility of the margin term in MMI/MPE based training. Ideally, we can turn on/off the margin term in the optimization problem.
2. Direct evaluation of the utility of an additional confidence term. Ideally, we improve over the best trained system by retraining the system with unsupervised labeled test data.
3. Direct evaluation of the amount of iterations and confidence-thresholds during optimization. In ASR, typically a low number of iterations is used in optimization, and confidence-thresholds are optimized on small subsets only. Here we allow for a high number of iterations on large datasets, and a detailed analysis of confidences in unsupervised model adaptation.
4. Evaluation on state-of-the-art systems. Ideally, we directly improve over the best discriminative system, e.g. conventional (i.e., without margin) MMI/MPE for handwriting recognition.

Due to the nature of the novel publicly available RWTH OCR[1] framework and databases, it can be assumed that most results can be transferred to ASR domains. Similar usage of features, lexica, and language models on smaller corpora allow for a detailed analysis of regularization, optimization iterations, as well as impact of confidence-thresholds.

The proposed approach takes advantage of the generalization bounds of large margin classifiers while keeping the efficient framework for conventional discriminative training. This allows us to directly evaluate the utility of the margin term for handwriting recognition. So, our approach combines the advantages of conventional training criteria and of large margin classifiers.

[1] http://www.hltpr.rwth-aachen.de/rwth-ocr/



(a) (b)

Figure 1 Two examples where each column shows the same Tunisian town name: large white-spaces (a) and elongation (b) occurs often in Arabic handwriting. Therefore an adequate modeling of white-spaces and state-transition penalties are important parts to be tuned in an HMM based Arabic handwriting recognition system.

2 System Overview

In offline handwriting recognition, we are searching for an unknown word sequence $w_1^N := w_1, \ldots, w_N$, for which the sequence of features $x_1^T := x_1, \ldots, x_T$ fits best to the trained models. We maximize the posterior probability $p(w_1^N|x_1^T)$ over all possible word sequences w_1^N with unknown number of words N. This is described by the Bayes' decision rule:

$$x_1^T \rightarrow \hat{w}_1^N(x_1^T) = \arg\max_{w_1^N} \left\{ p^\kappa(w_1^N) p(x_1^T|w_1^N) \right\} \quad (1)$$

with κ being a scaling exponent of the language model.

In this work, we use a writing variant model refinement [8] of our visual model

$$p(x_1^T|w_1^N) = \max_{v_1^N|w_1^N} \left\{ p_{\Lambda_\mathrm{v}}^\alpha(v_1^N|w_1^N) p_{\Lambda_\mathrm{e,t}}^\beta(x_1^T|v_1^N, w_1^N) \right\} \quad (2)$$

with v_1^N a sequence of unknown writing variants, α a scaling exponent of the writing variant probability depending on a parameter set Λ_v, and β a scaling exponent of the visual character model depending on a parameter set $\Lambda_\mathrm{e,t}$ for emission and transition model.

Especially in Arabic handwriting with its position-dependent shapes [23], large white-spaces can occur between isolated-, beginning-, and end-shaped characters (see Figure 1 (a)). As a specific set of characters is only connectable from the right side, such words have to be cut into pieces (Part of Arabic Word (PAW)). Due to ligatures and diacritics in Arabic handwriting, the same Arabic word can be written in several writing variants, depending on the writer's handwriting style.

During training, a corpus and lexicon with supervised writing variants instead of the commonly used unsupervised writing variants can be used, during

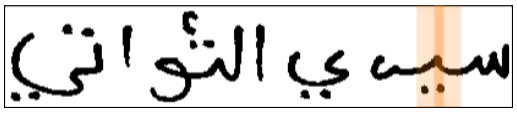


Figure 2 Right-to-left sliding PCA window over input images without any preprocessing for Arabic handwriting.

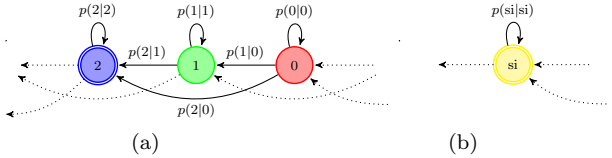


Figure 3 Different HMM topologies and transition probabilities are used for character models (a) and white-space models (b) in Arabic and Latin handwriting recognition.

decoding, the writing variants can only be used in an unsupervised manner. Obviously, the supervised writing variants in training can lead to better trained character models only if the training corpora have a high annotation quality. Usually, the probability $p(v|w)$ for a variant v of a word w is considered as uniformly distributed [7]. Here we use the count statistics as probability $p(v|w) = \frac{\mathsf{N}(v,w)}{\mathsf{N}(w)}$ where the writing variant counts $\mathsf{N}(v,w)$ and the word counts $\mathsf{N}(w)$ are estimated from the corresponding training corpora, and represent how often these events were observed. Note that $\sum_{v'} \frac{\mathsf{N}(v',w)}{\mathsf{N}(w)} = 1$. The scaling exponent α of the writing variant probability of Equation 2 can be adapted in the same way as it is done for the language model scale κ in Equation 1.

2.1 Feature Extraction

The aim of this work is to analyze the effect of discriminative training and the incorporation of a margin and a confidence term into the criteria. Therefore only few preprocessing steps commonly applied in handwriting recognition will be used: Deslanting as well as a size normalization are used to compensate for variations in Latin writing style as proposed by [20], no preprocessing will be used with Arabic handwritten data.

After an optional preprocessing of the input images, the images are scaled down to 16 pixel height while keeping their aspect ratio. We extract simple appearance-based image slice features x'_t at every time step $t = 1, \ldots, T$ which are augmented by their spatial derivatives in horizontal direction $\Delta = x'_t - x'_{t-1}$. Note that many systems divide the sliding window itself into several sub-windows and extract different features within each of the sub-windows [2, 20, 30, 39]

In order to incorporate temporal and spatial context into the features, we concatenate 7 consecutive features in a sliding window with maximum overlap, which are later reduced by a PCA transformation matrix to a feature vector x_t of dimension 30 (see Figure 2).

2.2 Visual Modeling

Our hidden Markov model (HMM) based handwriting recognition system is Viterbi trained using the maximum-likelihood (ML) training criterion and a lexicon with multiple writing variants as proposed in [7,8].

Each character is modeled by a multi-state left-to-right HMM with skip transitions and separate Gaussian mixture models (GMMs). The parameters of all GMMs are estimated with the ML principle using an expectation maximization (EM) algorithm, and to increase the number of densities in the mixture densities, successive splitting of the mixture densities is applied. Different HMM topologies and transition probabilities are used for character models (cf. Figure 3(a)) and white-space models (cf. Figure 3(b)) in Arabic and Latin handwriting recognition, where the white-space model itself is always modelled by a single GMM in all systems.

Arabic handwriting. Depending on the position of the character in an Arabic word, most of the 28 characters can have up to 4 different shapes [23]. Here we use position-dependent character models to model the different presentation forms, and due to ligatures, a total of 120 character models and one white-space model have to be estimated in training (see Section 4). Each character model in our Arabic handwriting recognition base system is modeled by a 3-state left-to-right HMM with three separate GMMs. The position-dependent character-based model of our ML trained baseline system includes 361 mixtures with 36k Gaussian densities (with up to 128 densities per mixture) with globally pooled diagonal variances. Additionally, a large stretching of long drawn-out characters occurs often in Arabic handwriting (see Figure 1 (b)). Therefore, we use very low loop penalties but higher skip penalties for our HMM state transitions (see Figure 3 (a)).

Latin handwriting. The Latin handwriting is one of the most common handwriting systems worldwide. English handwriting uses the alphabets 26 basic characters. As each letter can be written in lower- and uppercase, and capitalized or cursive writing, and additionally symbols for punctuations are used in the IAM database (see Section 4), 78 character models and one blank model have to be estimated in our ML trained baseline system, where each character model is modeled by a 10-state left-to-right HMM with five separate GMMs, resulting in 391 mixtures with 25k Gaussian

densities (with up to 128 densities per mixture) after ML training and globally pooled diagonal variances.

2.3 Discriminative Training: Incorporation of the Margin and Confidence Term

In this work, we use a discriminative training approach based on the Maximum Mutual Information (MMI) and Minimum Phone Error (MPE) criteria as presented in [16, 17, 15]. In addition to the novel confidence-based extension of the margin-based MMI training presented in [6], the confidence concept has been incorporated in the margin-based MPE criterion in this work. In the following, we give a brief summary.

The two-dimensional representation of a handwritten image is turned into a string representation $X = x_1, \ldots, x_T$ where x_t is a fixed-length array assigned to each column in the image (see Section 2.1 for further details). The word sequence $W = w_1, \ldots, w_N$ is represented by a character string.

Assume the joint probability $p_\Lambda(X, W)$ of the features X and the symbol string W. The model parameters are indicated by Λ. The training set consists of $r = 1, \ldots, R$ labeled sentences, $(X_r, W_r)_{r=1,\ldots,R}$. According to Bayes rule, the joint probability $p_\Lambda(X, W)$ induces the posterior

$$p_{\Lambda,\gamma}(W|X) = \frac{p_\Lambda(X, W)^\gamma}{\sum_V p_\Lambda(X, V)^\gamma}. \quad (3)$$

The likelihoods are scaled with some factor $\gamma > 0$, which is a common trick in speech recognition to scale them to the "real" posteriors [17]. The approximation level γ is an additional parameter to control the smoothness of the criterion.

Let $p_\Lambda(X, W)$ be the joint probability and $L[p_\Lambda(X_r, \cdot), W_r]$ a loss function for each training sample r with $\cdot$ representing all possible hypotheses W for a given lexicon, and W_r representing the correct transcription of X_r. The general optimization problem is now formulated as a minimization of the total loss function:

$$\hat{\Lambda} = \arg\min_\Lambda \{C||\Lambda - \Lambda_0||_2^2 + \sum_{r=1}^{R} L[p_\Lambda(X_r, \cdot), W_r]\} \quad (4)$$

and includes an ℓ_2 regularization term $||\Lambda - \Lambda_0||_2^2$ (i.e. a prior over the model parameters), where the constant C is used to balance the regularization term and the loss term including the log-posteriors. Here, the regularization is replaced by I-smoothing [37], which is a useful technique to make MMI/MPE training converge without over-training, and where the parameter prior is centered for initialization at a reasonable ML trained model Λ_0 (see Section 2.2).

2.3.1 Margin-Based Maximum Mutual Information

In automatic speech recognition (ASR), maximum mutual information (MMI) commonly refers to the maximum likelihood (ML) for the class posteriors. For MMI, the loss function to be minimized is described by:

$$L^{(\mathrm{MMI})}[p_\Lambda(X_r, \cdot), W_r] = -\log \frac{p_\Lambda(X_r, W_r)^\gamma}{\sum_V p_\Lambda(X_r, V)^\gamma}. \quad (5)$$

This criterion has proven to perform reasonably as long as the error rate on the training data is not too low, i.e., generalization is not an issue.

The margin-based MMI loss function (M-MMI) to be minimized is described by:

$$L_\rho^{(\text{M-MMI})}[p_\Lambda(X_r, \cdot), W_r] = -\log \frac{[p_\Lambda(X_r, W_r) \exp(-\rho A(W_r, W_r))]^\gamma}{\sum_V [p_\Lambda(X_r, V) \exp(-\rho A(V, W_r))]^\gamma}, \quad (6)$$

which has an additional margin-term including the accuracy $A(\cdot, W_r)$ being maximal for the correct transcription $W = W_r$. Note that the additional term can be interpreted as if we had introduced a new posterior distribution. In a simplified view, we interpret this as a pseudo-posterior probability which is modified by a margin term.

Compared with the true-posterior in Equation 3, the margin pseudo-posterior includes the margin term $\exp(-\rho A(V, W_r))$, which is based on the string accuracy $A(V, W_r)$ between the two strings V, W_r. The accuracy counts the number of matching symbols of V, W_r and will be approximated for efficiency reasons (see Section 2.3.3) by the approximate word accuracy [35].

As explained in [17], the accuracy is generally scaled with some $\rho > 0$, and this term weighs up the likelihoods of the competing hypotheses compared with the correct hypothesis [36]. On the contrary, this term can be equally interpreted as a margin term.

This margin term can be interpreted as an additional observation dependent prior, weakening the true prior [18]. Moreover, this training criterion is identical with the SVM optimization problem for $\gamma \to \infty$ and log-linear models [16]. Keep in mind that Gaussian HMMs with globally pooled variances are equivalent to a log-linear model with first order features only [14]. The loss functions for MMI and M-MMI are compared with the hinge loss function in Figure 4. The example is given for a binary classification problem with single observations (i.e. no symbol strings). The loss function is plotted against the log-ratio of the posterior of the correct class W_n to the posterior of the competing class $\bar{W}_n$

$$d = \log\left(\frac{p_\Lambda(X_n, W_n)}{p_\Lambda(X_n, \bar{W}_n)}\right) \quad (7)$$

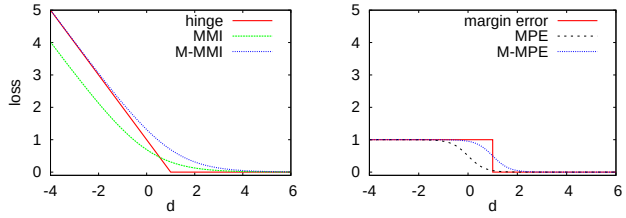


Figure 4 Comparison of loss functions for a binary classification problem with d as defined in Equation 7. Left: comparison of MMI and M-MMI loss functions with the hinge loss function. Right: comparison of MPE and M-MPE loss functions with the margin error. Note that the margin term shifts the loss function such that the inflection point is at $d = 1$ and not $d = 0$.

for $\gamma = 1$, $\rho = 1$, and $A(V, W) = \delta(V, W)$. MMI and M-MMI differ by an offset $d = 1$, and M-MMI is a smooth approximation to the hinge loss function (for more details cf. [16, 17]).

2.3.2 Margin-Based Minimum Phone Error

The Minimum Phone Error (MPE) criterion is defined as the (regularized) posterior risk based on the error function $E(V, W)$, which is probably the training criterion of choice in Large Vocabulary Continuous Speech Recognition (LVCSR). For MPE, the loss function to be minimized is described by:

$$L^{(\mathrm{MPE})}[p_\Lambda(X_r, \cdot), W_r] = \sum_{W \in \cdot} E(W, W_r) \frac{p_\Lambda(X_r, W_r)^\gamma}{\sum_V p_\Lambda(X_r, V)^\gamma}, \tag{8}$$

which is based on the error function $E(V, W)$ like for example the approximate phone error [35]. In OCR, a phoneme unit usually corresponds to a character if words are modeled by character sequences.

Analogously, the margin-based MPE loss function (M-MPE) to be minimized is described by:

$$L_\rho^{(\text{M-MPE})}[p_\Lambda(X_r, \cdot), W_r] = \sum_{W \in \cdot} E(W, W_r) \frac{[p_\Lambda(X_r, W_r) \exp(-\rho A(W, W_r))]^\gamma}{\sum_V [p_\Lambda(X_r, V) \exp(-\rho A(V, W_r))]^\gamma}, \tag{9}$$

It should be noted that due to the relation $E(W, W_r) = |W_r| - A(W, W_r)$ where $|W_r|$ denotes the number of symbols in the reference string W_r, the error $E(W, W_r)$ and the accuracy $A(W, W_r)$ can be equally used in Equation 8 and Equation 9. The accuracy for MPE and for the margin term do not need to be the same quantity [15].

Again, the loss functions for MPE and M-MPE are compared for a binary classification problem with single observations (for $E(V, W) = 1 - \delta(V, W)$, $A(V, W) = \delta(V, W)$, $\gamma = 3$, $\rho = 1$) in Figure 4. The illustration shows that M-MPE is a horizontally shifted version of MPE, while M-MPE approximating the margin error. Finally, it should be pointed out that other posterior-based training criteria (e.g. MCE as used in [5]) can be modified in an analogous way to incorporate a margin term (for more details cf. [16, 17]).

Table 1 Empirical optimization of the I-smoothing regularization constant C for the IAM line recognition task: the Word Error Rate (WER) and Character Error Rate (CER) results after five Rprop optimization iterations.

	WER [%]		CER [%]	
Regularization constant C	Devel	Test	Devel	Test
0.001	33.25	39.43	10.68	15.64
0.01	33.17	39.40	10.63	15.66
0.1	33.26	39.44	10.70	15.67
1.0	33.14	39.42	10.64	15.63
10.0	33.12	39.44	10.64	15.67

2.3.3 Optimization

In [16] it is shown that the objective function $\mathcal{F}_\gamma^{(\mathrm{MMI})}(\Lambda)$ converges pointwise to the SVM optimization problem using the hinge loss function for $\gamma \to \infty$, similar to [42]. In other words, $\mathcal{F}_\gamma^{(\text{M-MMI})}(\Lambda)$ is a smooth approximation to an SVM with hinge loss function which can be iteratively optimized with standard gradient-based optimization techniques like Rprop [16, 42].

In this work, the regularization constant C, the approximation level γ, and the margin scale ρ are chosen beforehand and then kept fixed during the complete optimization. Note that the regularization constant C and the margin scale ρ are not completely independent of each other. Here, we kept the margin scale ρ fixed and tuned the regularization constant C (see Table 1). Previous experiments in ASR have suggested that the performance is rather insensitive to the specific choice of the margin [16], and the results in Table 1 furthermore suggest that if the baseline error rate is relatively high the choice of the I-smoothing constant C has less impact in an Rprop based optimization than in an Extendend Baum Welch (EBW) environment [37]. An I-smoothing regularization constant $C = 1.0$ is used in all results presented in Section 4.

In large-vocabulary handwriting recognition, word lattices restricting the search space are used to make the summation over all competing hypotheses (i.e. sums over W) efficient. The exact accuracy on character or word level cannot be computed efficiently due to the Levenshtein alignments in general, although

feasible under certain conditions as shown in [15]. Thus, the approximate phone/word accuracy known from MPE/MWE [35] is used for the margin instead. With this choice of accuracy, the margin term can be represented as an additional layer in the common word lattices such that efficient training is possible. More details about the transducer-based implementation used in this work can be found in [15].

As in ASR, where typically a weak unigram language model is used for discriminative training [40,41], we use a unigram language model in our proposed discriminative training criteria.

2.3.4 Confidences for Unsupervised Discriminative Model Adaptation

Sentence or word confidences can be incorporated into the training criterion by simply weighing the segments with the respective confidence. This is, however, not possible for state-based confidences. Instead of rejecting an entire sentence or word the system can use state confidence scores to select state-dependent data in an unsupervised manner. State confidence scores are obtained from computing arc posteriors from the lattice output from a previous decoder pass.

Rprop is a gradient-based optimization algorithm. The gradient of the training criterion under consideration can be represented in terms of the state posteriors $p_{rt}(s|x_1^{T_r})$. These posteriors are obtained by marginalization and normalization of the joint probabilities $p_\Lambda(x_1^{T_r}, s_1^T, w_1^{N_r})$ over all state sequences through state s at frame t. These quantities can be calculated efficiently by recursion, cf. forward/backward probabilities. Then, the state-based confidences are incorporated by multiplying the posteriors with the respective confidence before the accumulation. In summary, each frame t contributes $conf(t) \cdot p_{rt}(s|x_1^{T_r}) \cdot x_t$ to the accumulator acc_s of state s.

Another way to describe the incorporation of the confidence term is from a system point of view. The accumulator acc_s of state s can be described by

$$\mathrm{acc}_s = \sum_{r=1}^{R} \sum_{t=1}^{T_r} \omega_{r,s,t} \cdot x_t,$$

where the weight $\omega_{r,s,t}$, which corresponds to $\delta(s_t, s)$ in ML training i.e. one or zero, is replaced for the proposed M-MMI-conf / M-MPE-conf criteria (with $\rho \neq 0$) by the corresponding margin pseudo-posterior. With the additional confidence term for the proposed M-MMI-conf criterion (cf. Equation 6), the new weight can be described as follows:

$$\omega_{r,s,t} := \underbrace{\frac{\sum_{s_1^{T_r}:s_t=s} [p(x_1^{T_r}|s_1^{T_r})p(s_1^{T_r})p(W_r)\, e^{-\rho A(W_r,W_r)}]^\gamma}{\sum_{V} \sum_{s_1^{T_r}:s_t=s} [p(x_1^{T_r}|s_1^{T_r})p(s_1^{T_r})p(V)\, \underbrace{e^{-\rho A(V,W_r)}}_{\text{margin}}]^\gamma}}_{\text{posterior}} \cdot \underbrace{\delta(c_{r,s,t} \geq \tau_c)}_{\text{confidence selection}} \quad (10)$$

Here, the selector function $\delta(c_{r,s,t} \geq \tau_c)$ with the threshold parameter τ_c controls the amount of adaptation data. The M-MPE-conf criterion can be defined in a similar manner. Note that due to the quality of the confidence metric, thresholding the confidence scores after feature selection can often result in an improved accuracy, as reported in [12]. On the one hand, the experimental results for word-confidences in Figure 12 and state-based confidences in Figure 16 suggest that the confidences are helpful, but on the other hand that the threshold itself has little impact due the proposed M-MMI-conf / M-MPE-conf approaches, which are inherently robust against outliers.

Analogously, the weight $\omega_{r,s,t}$ would correspond to the true posterior (cf. Equation 3) in an MMI-conf / MPE-conf criterion. Note that in informal experiments these criteria lead to no robust improvements, i.e. only the combination of margin *and* confidences makes the proposed approaches robust against outliers.

3 Decoding Architecture

The recognition is performed in two passes. System 1 performs the initial and independent recognition pass using the discriminatively trained models. The output is required for the unsupervised text dependent model adaptation in the next step.

For unsupervised adaptation, at test time, the conditioning state sequence is derived from a prior recognition pass. Although the prior transcript in that case contains errors, adapting on that transcript disregarding that fact generally still results in accuracy improvements [12].

The model adaptation in the second pass is performed by discriminatively training a System 2 on the text output of the first-pass recognition system. Additionally, the confidence-alignments generated during the first-pass decoding can be used on a sentence-, word-, or state-level to exclude the corresponding features from the discriminative training process for unsupervised model adaptation.

Out-of-vocabulary (OOV) words are also meant to be harmful for adaptation [34] but even when a word is

wrong, the pronunciation or most of the pronunciation can still be correct, suggesting that a state-based and confidence-based adaptation should be favored in such cases.

3.1 Word Confidences

As we are dealing with isolated word recognition on the IFN/ENIT database, the sentence and word confidences are identical. The segments to be used in the second-pass system are first thresholded on a *word-level* by their word confidences: only complete word *segments* aligned with a high confidence by the first-pass system are used for model adaptation using discriminative training.

3.2 State Confidences

Instead of rejecting an entire sentence or word, the system can use state confidence scores to select state-dependent data (cf. Section 2.3.4). State confidence scores are obtained from computing arc posteriors from the lattice output of the decoder. The arc posterior is the fraction of the probability mass of the paths that contain the arc from the mass that is represented by all paths in the lattice. The posterior probabilities can be computed efficiently using the forward-backward algorithm as, for example, described in [21]. Then, the word frames to be used in the second-pass system are first thresholded on a *state-level* by their state confidences: only word *frames* aligned with a high confidence by the first-pass system, are used for model adaptation using discriminative M-MMI-conf/M-MPE-conf training (see Section 2.3).

An example for a word-graph and the corresponding 1-best state alignment is given in Figure 5: during the decoding, the ten feature frames (the squares) can be aligned to different words (long arcs) and their states. In this example, the word-confidence of the 1-best alignment is $c = 0.7$ (upper arc). The corresponding state-confidences are calculated by accumulating state-wise over all competing word alignments (lower arcs), i.e. the state-confidence of the 1-best alignment's fourth state would stay 0.7 as this state is skipped in all other competing alignments, all other state-confidences would sum up to 1.0.

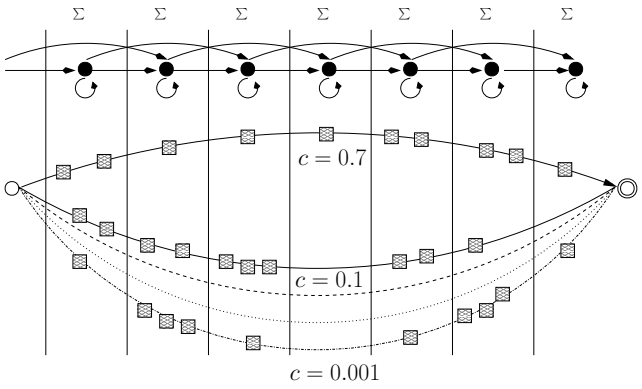


Figure 5 Example for a word-graph and the corresponding 1-best state alignment: word-confidence of the 1-best alignment is $c = 0.7$. The corresponding state-confidences are calculated by accumulating state-wise over all other word alignments

4 Experimental Results

The proposed approach is applied to isolated Arabic handwriting and continuous English handwriting. The experiments for isolated word recognition are conducted on the IFN/ENIT database [32] using a closed lexicon, experiments for continuous sentence recognition on the IAM database [27] using a large-vocabulary lexicon and additional external language model resources as proposed in [3].

The IFN/ENIT database is divided into four training folds with an additional fold for testing [25]. The current database version (v2.0p1e) contains a total of 32492 Arabic words handwritten by about 1000 writers, and has a vocabulary size of 937 Tunisian town names. Here, we follow the same evaluation protocol as for the ICDAR 2005, 2007, and 2009 competitions [24] (see Figure 6). The corpus statistics for the different folds can be found in Table 2.

The IAM database was introduced by [27] in 2002 and contains a total number of 1,539 pages with 5,685 sentences in 9,862 lines. All words are build using only 79 different symbols which consist of both upper- and lowercase characters, punctuation, quotation marks, a special symbol for crossed out words, and a white-space model (cf. Section 2.2). A comparison of the predefined training, testing and evaluation folds is given in Table 3. Here we focus on the large-vocabulary line recognition task, which is one of the four tasks provided with the database. For the large-vocabulary recognition task we use as proposed in [3] the three additional text corpora Lancaster-Oslo-Bergen, Brown and Wellington (LBW) to estimate our language models and lexica. Note that the IAM validation/test lines were excluded from the Lancaster-Oslo-Bergen (LOB) corpus.

4.1 First Pass Decoding

In this section we compare our ML trained baseline systems (cf. Section 2.2 for visual model details) to our

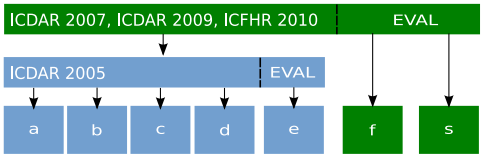


Figure 6 IFN/ENIT corpora splits used for training and evaluation in Arabic handwriting recognition competitions organized at ICDAR 2005, 2007, and 2009, and ICFHR 2010.

Table 2 Corpus statistics for the IFN/ENIT Arabic handwriting sub-corpora.

Folds	#Observations [k]			
	Writers	Words	Characters	Frames
a	0.1	6.5	85.2	452
b	0.1	6.7	89.9	459
c	0.1	6.5	88.6	452
d	0.1	6.7	88.4	451
e	0.5	6.0	78.1	404
f	n.a.	8.6	64.7	n.a.
s	n.a.	1.5	11.9	n.a.

Table 3 Corpus statistics for the IAM Latin handwriting corpus using a 50k lexicon

	Train	Devel	Eval	LM
words	53,884	8,717	25,472	3,363,402
chars	219,749	31,724	96,637	13,871,031
lines	6,161	920	2,781	164,944
writers	283	57	162	-
OOV rate	1.07%	3.94%	3.42%	1.87%

discriminatively trained systems using the MMI and MPE criteria and their margin-based extensions. The discriminative training is initialized with the respective ML trained baseline model and iteratively optimized using the Rprop algorithm (cf. Section 2.3).

Isolated Word Recognition. For isolated Arabic word recognition, we compare our ML trained baseline system with MMI/M-MMI criteria only.

In general, the number of Rprop iterations and the choice of the regularization constant C have to be chosen carefully (cf. optimization Table 1 in Section 2.3), and were empirically optimized in informal experiments to 30 Rprop iterations and $C = 1.0$ (cf. detailed Rprop iteration analysis and convergence without overtraining in Figure 8 and Figure 9).

The results in Table 4 show that the discriminatively trained models clearly outperform the ML trained baseline models, especially the models trained with the additional margin term. The strong decrease in word error rate (WER) for experiment setup *abd-c* might be due to the training data being separable for the given configurations, whereas the strong improvement

Table 4 Comparison of ML trained baseline systems, and discriminatively trained systems using MMI and M-MMI criteria after 30 Rprop iterations on the IFN/ENIT database.

Train	Test	WER[%]		
		ML	MMI	M-MMI
abc	d	10.88	10.59	**8.94**
abd	c	11.50	10.58	**2.66**
acd	b	10.97	10.43	**8.64**
bcd	a	12.19	11.41	**9.59**
abcd	e	21.86	21.00	**19.51**
abcde	e	11.14	2.32	**2.95**

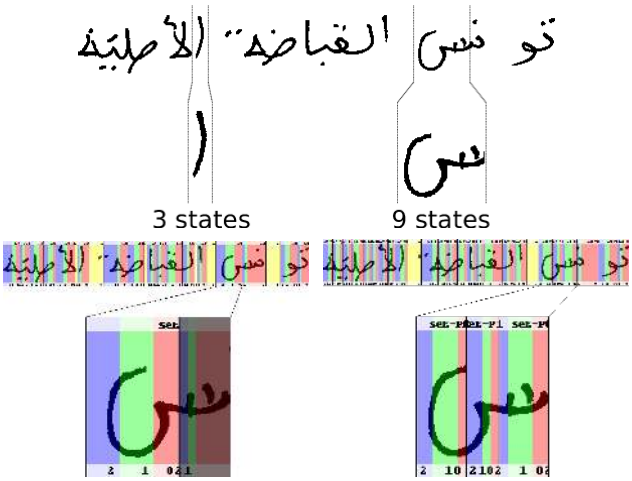


Figure 7 Top: more complex characters should be represented by more states. Bottom: after the GDL, frames previously aligned to a wrong neighboring character model (left, black shaded) are aligned to the correct character model (right, three sub-glyphs).

for experiment *abcde-e* was expected because of the test set *e* being part of the training data.

In the following experiments, we additionally use a glyph dependent model length estimation (GDL) as described in [7, 8], resulting in an ML trained baseline model with 637 mixtures and 48k densities (cf. Section 2.2). The necessity of this model length estimation is visualized in Figure 7, where we use R-G-B background colors for the 0-1-2 HMM states (also cf. Figure 3), respectively, from right-to-left: the bottom row images visualize an alignment of our baseline system (left) in comparison to the proposed GDL system (right).

By estimating glyph dependent model lengths, the overall mean of character length changed from 7.89 pixels (i.e. 2.66 pixels/state) to 6.18 pixels (i.e. 2.06 pixels/state) when downscaling the images to 16 pixels height while keeping their aspect-ratio. Thus every state of an GDL character model has to cover less pixels due to the relative reduction of approx. 20% pixels.

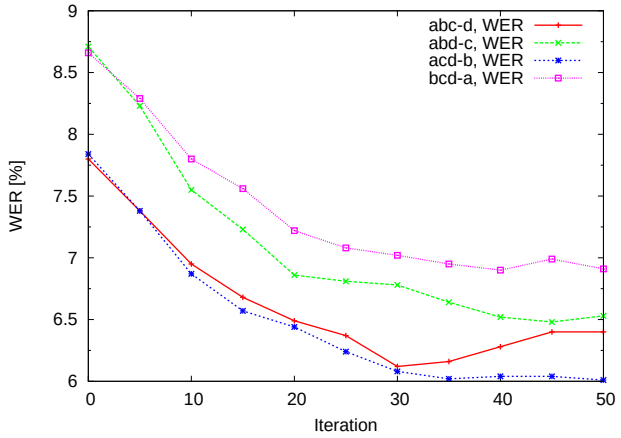


Figure 8 Decreasing word error rates (WER) for all different training folds of the IFN/ENIT database over M-MMI Rprop iterations (baseline with model length estimation).

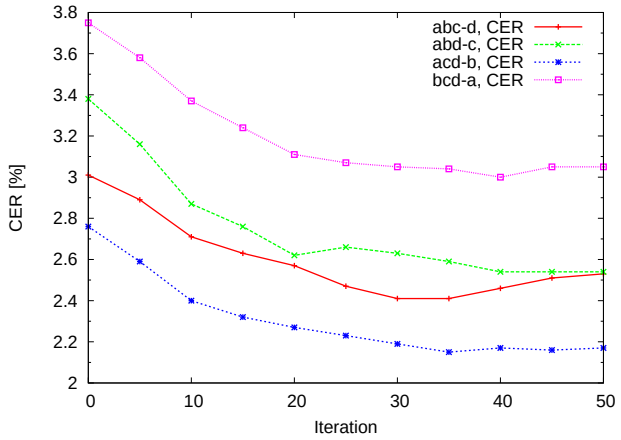


Figure 9 Decreasing character error rates (CER) for all different training folds of the IFN/ENIT database over M-MMI Rprop iterations (baseline with model length estimation).

In Figure 8 and Figure 9, detailed WER and character error rate (CER) plots over M-MMI training iterations are shown, respectively, with Figure 10 showing a combined WER/CER plot over M-MMI training iterations on the evaluation setup *abcd-e*. It can be observed that both WER and CER are smoothly and almost continuously decreasing with every Rprop iteration, and that about 30 Rprop iterations are optimal for the considered datasets.

Continuous Large-Vocabulary Line Recognition. For the large-vocabulary line recognition task on the IAM database, our system uses a Kneser-Ney smoothed trigram language model [22] trained on the LBW text corpora (cf. Section 2.2 for visual model details and cf. [19] for a detailed description of the ML baseline system). Note that for discriminative training a weakened unigram language model is used as explained in Section 2.3. The language model weighting factor $\kappa = 25$ (cf. Equation 1) and the word insertion penalty were determined empirically on the validation set using the ML trained models. Again, the discriminative training is initialized with the respective ML trained baseline model and iteratively optimized using the Rprop algorithm (cf. Section 2.3).

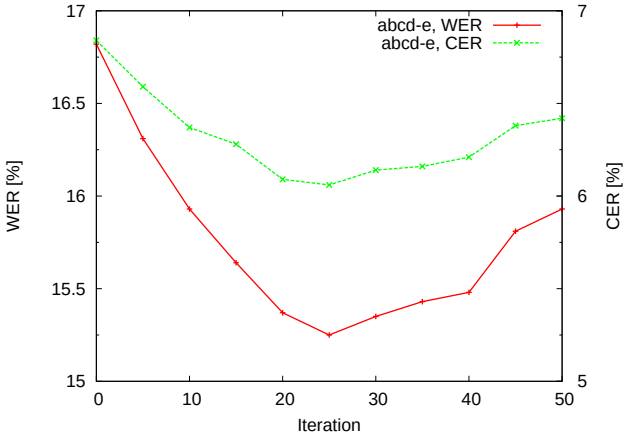


Figure 10 Evaluation of the proposed M-MMI training on the IFN/ENIT evaluation setup *abcd-e* over M-MMI Rprop iterations (baseline with model length estimation).

Table 5 Word error rate (WER) and character error rate (CER) results for the IAM line recognition task after 100 Rprop iterations.

		WER [%]		CER [%]	
Criterion	Lexicon	Devel	Test	Devel	Test
ML [19]	20k	34.6	41.5	8.9	11.0
	50k	31.9	39.0	8.4	11.8
MMI	50k	25.9	31.8	7.6	12.0
M-MMI	50k	25.8	31.6	7.6	11.8
MPE	50k	24.4	30.3	**6.7**	11.1
M-MPE	50k	**24.3**	**30.1**	6.9	**10.9**

Results for discriminative training in comparison to our ML trained baseline system are shown in Table 5. The lexicon size of 50k has been roughly optimized on the ML trained baseline system and used for all further experiments.

The results in Table 5 were obtained after 100 Rprop iterations, as shown for M-MMI/M-MPE in Figure 11. Note the smooth decrease of both WER and CER after every iteration. Similar figures are obtained with the unmodified MMI/MPE criteria. It can be observed that the margin modified criteria always slightly outperform their corresponding standard criteria, and that the MPE based criteria outperform the MMI based criteria, especially w.r.t. CER. However, the results in Table 5 support the hypothesis that the effect of the margin on such highly competitive large-vocabulary systems used for discriminative training is sometimes marginal [17].

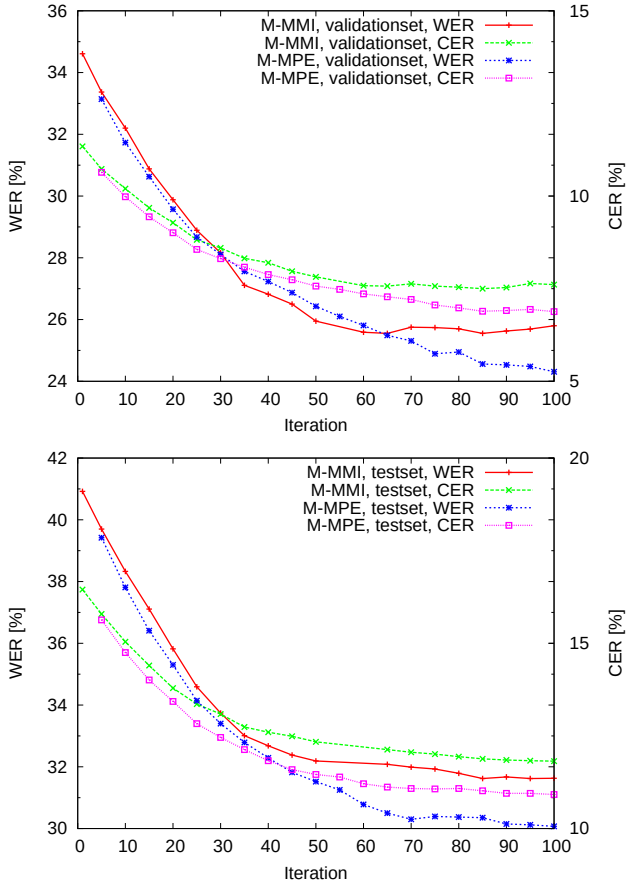


Figure 11 M-MMI/M-MPE training on the IAM database over 100 Rprop iterations with a smooth decrease in word error rate (WER, left axis) and character error rate (CER, right axis).

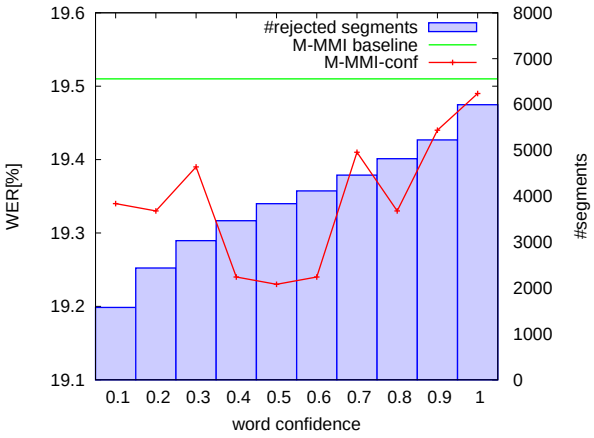


Figure 12 Results for word-confidence based M-MMI-conf training on the IFN/ENIT database using different confidence thresholds and their corresponding number of rejected segments (baseline without model length estimation).

4.2 Second Pass Decoding and Unsupervised Model Adaptation

In this section we evaluate our discriminative training for unsupervised model adaptation during a second pass decoding step.

In a first experiment we used the complete first-pass output of the M-MMI system for an unsupervised model adaptation. The results in Table 6 show that the M-MMI based unsupervised adaptation without confidences cannot improve the system accuracy. With every Rprop iteration, the system is even more biased by the relatively large amount of wrong transcriptions in the adaptation corpus.

The discriminative M-MMI-conf training is initialized with the respective M-MMI trained model and iteratively optimized using the Rprop algorithm (cf. Section 2.3). Using the word-confidences for M-MMI-conf based model adaptation of our first-pass alignment to reject complete word segments (i.e. feature sequences X_1^T) from the unsupervised adaptation corpus, the results in Table 6 show a slight improvement only in comparison to the M-MMI trained system. Figure 12 shows the resulting WER for different confidence threshold values and the corresponding number of rejected segments. For a confidence threshold of $c = 0.5$, more than 60% of the 6033 segments of set e are rejected from the unsupervised adaptation corpus, resulting in a relatively small amount of adaptation data.

Using the state-confidences for M-MMI-conf based model adaptation of our first-pass alignment to decrease the contribution of single frames (i.e. features X_t) during the iterative M-MMI-conf optimization process (cf. optimization in Section 2.3), the number of features for model adaptation is reduced by approximately 5% for a confidence threshold of $c_{\text{threshold}} = 0.5$: 375 446 frames of 396 416 frames extracted from the 6033 test segments are considered during the optimization, only 20 970 frames are rejected based on confidence thresholding (cf. also Figure 5). Note that also the CER is decreased to 6.49%.

Interestingly, the supervised adaptation on test set e, where only the correct transcriptions of set e are used for an adaptation of the model trained using set *abcd*, can again decrease the WER of the system down to 2.06%, which is even better than an M-MMI optimization on the full training set *abcde* (cf. Table 4).

In Figure 13 and Figure 14, detailed WER and CER plots over M-MMI-conf training iterations are shown, respectively, with Figure 15 showing a combined WER/CER plot over M-MMI-conf training iterations on the evaluation setup *abcd-e* (cf. initialization plots). In all cases, we estimated the state-confidences on the first pass output using the M-MMI trained models. It can be observed that both WER and CER are slightly decreasing with every Rprop iteration, and

Table 6 Results for M-MMI-conf model adaptation on the evaluation setup *abcd-e* of the IFN/ENIT database after 30 Rprop iterations (baseline without model length estimation).

Training/Adaptation	WER[%]	CER[%]
ML	21.86	8.11
M-MMI	19.51	7.00
+ unsupervised adaptation	20.11	7.34
+ supervised adaptation	2.06	0.77
M-MMI-conf (word-confidences)	19.23	7.02
M-MMI-conf (state-confidences)	**17.75**	**6.49**

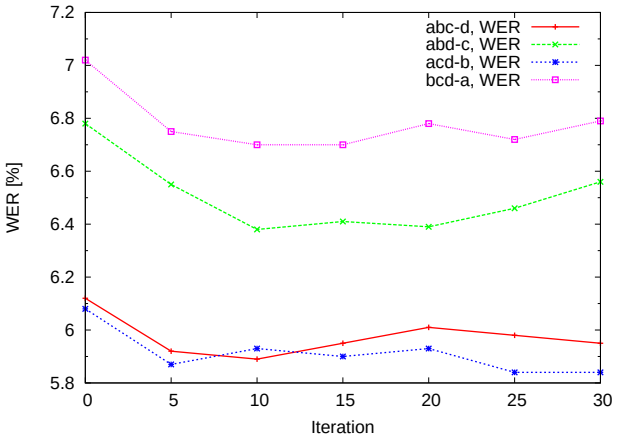


Figure 13 Decreasing word error rates (WER) for all different training folds on the IFN/ENIT database over confidence-based M-MMI-conf Rprop iterations (baseline with model length estimation).

Table 7 Results for confidence-based M-MMI-conf model adaptation after 15 Rprop iterations on the IFN/ENIT database using glyph dependent lengths (GDL), and margin-based M-MMI criterion after 30 Rprop iterations.

Train	Test	WER[%]				
		1st pass				2nd pass
		ML	GDL	+MMI	+M-MMI	M-MMI-conf
abc	d	10.9	7.8	7.4	**6.1**	**6.0**
abd	c	11.5	8.8	8.2	**6.8**	**6.4**
acd	b	11.0	7.8	7.6	**6.1**	**5.8**
bcd	a	12.2	8.7	8.4	**7.0**	**6.8**
abcd	e	21.9	16.8	16.4	**15.4**	**14.6**

that between 10 and 15 Rprop iterations are optimal for the considered small and unsupervised labeled test datasets.

Table 7 shows the final results of our Arabic handwriting recognition system with additional glyph dependent model length estimation (GDL) as described in [6]. Again, the WER of the GDL based system can be decreased by our proposed M-MMI training during both decoding passes down to 14.55%.

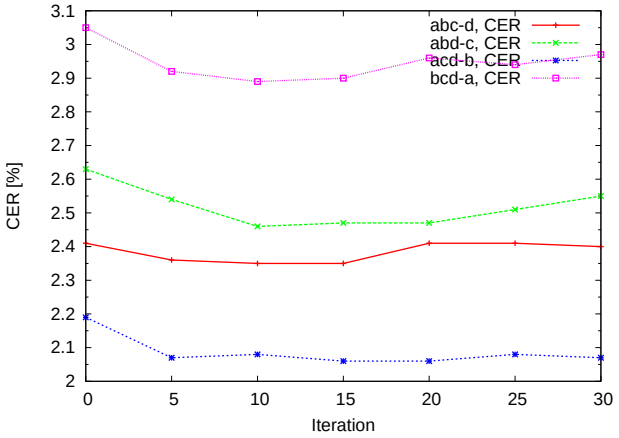


Figure 14 Decreasing character error rates (CER) for all different training folds on the IFN/ENIT database over confidence-based M-MMI-conf Rprop iterations (baseline with model length estimation).

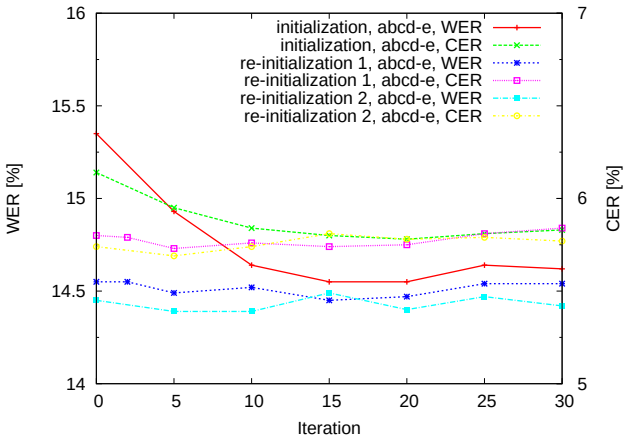


Figure 15 Evaluation of iterative M-MMI-conf model adaption: text transcriptions are updated in an unsupervised manner after 15 Rprop iterations. The performance remains robust even after several re-initializations.

Due to the robustness of the confidence- and margin-based M-MMI-conf criterion against outliers, the proposed unsupervised and text dependent model adaptation can be applied in an iterative manner by a re-initialization of the text transcriptions. In Figure 15, we re-initialize 2 times the model adaptation process after 15 Rprop iterations. The results in Figure 15 show the robustness of our approach, leading to an improved WER of 14.39%.

For the confidence-based unsupervised model adaptation approaches on the IAM database we also measured the performance after 15 Rprop iterations. The results in Figure 16 suggest that the often mentioned stronger robustness of the MPE criterion w.r.t. outliers than the MMI criterion [17] cannot be confirmed for continuous handwriting recognition within the proposed confidence-based M-MMI-conf and M-MPE-conf

criteria, as both approaches achieve a similar performance: M-MMI-conf decreases the error rates from 31.63% WER / 11.82% CER down to 29.02% WER / 10.52% CER, i.e. a relative improvement in WER of 8%, whereas M-MPE-conf decreases from 30.07% WER / 10.92% CER down to 29.23% WER / 10.33% CER, i.e. a relative improvement in WER of 2%. Note that in both cases the best unsupervised *transcriptions* of the unknown validation and test data from the M-MPE model has been used, but that the confidence-based model adaptation has been applied to the corresponding un-adapted *models*, i.e. M-MMI-conf to adapt the M-MMI trained model, and M-MPE-conf to adapt the M-MPE trained. This might explain the higher relative improvement in case of M-MMI-conf model adaptation. Also note that, as expected, the CER is lower for M-MPE-conf than for M-MMI-conf.

The number of rejected frames in Figure 16 is reported in both cases for the testset only, where a confidence-based reduction by approximately 5% of the number of features for model adaptation is again a good choice.

It can be observed that both criteria are robust against outliers, as the confidence-threshold, although helpful for values $c_{\text{threshold}} \leq 0.9$ (cf. Equation 10), has only a small impact on the overall performance of the model adaptation procedures. Interestingly, and opposed to the results for isolated word recognition in Table 6, the performance is also improved if all data is used in M-MMI-conf / M-MPE-conf for model adaptation. M-MMI-conf in Figure 16 seems to be less susceptible to unsuitable confidence-threshold and can therefore be considered the better unsupervised model adaptation approach if WER as evaluation criterion is important, otherwise M-MPE-conf might be the method of choice if CER as evaluation criterion is important. In particular, the achieved 29% WER for single and purely HMM based system is one of the best known word error rates for this task (cf. Section 4.4).

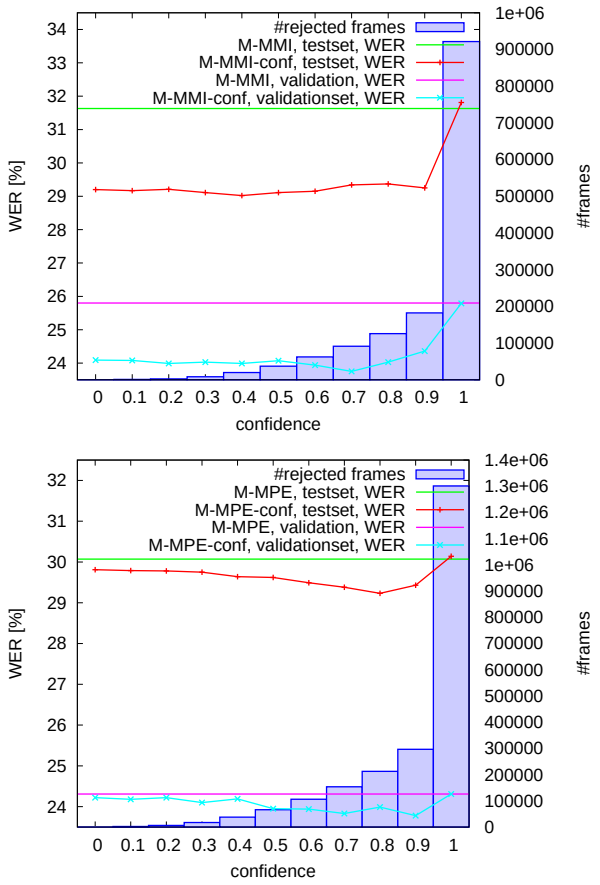
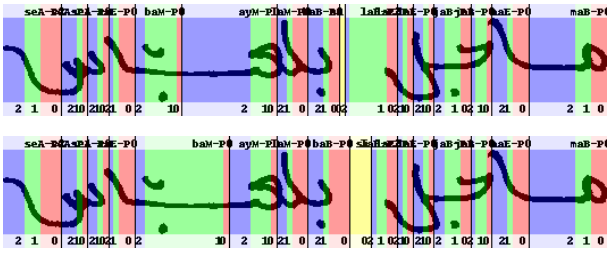


Figure 16 Comparison of the proposed M-MMI-conf and M-MPE-conf model adaption approaches: both approaches are robust against outliers, as the confidence-threshold, although helpful, has only a small impact on the overall performance

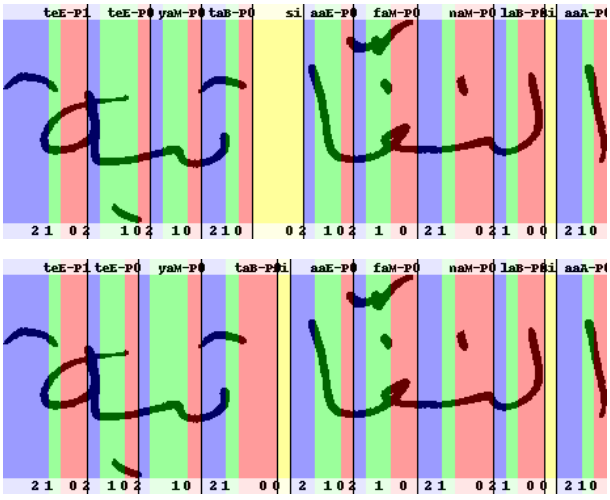
4.3 Visual Inspections

The visualizations in Figure 17 and Figure 18 show training alignments of Arabic words to their corresponding HMM states. The upper rows show the alignment to the ML trained model, the lower rows to the M-MMI trained models. We use R-G-B background colors for the 0-1-2 HMM states, respectively, from right-to-left. The position-dependent character model names (cf. Section 2.2) are written in the upper line, where the white-space models are annotated by 'si' for 'silence'; the state numbers are written in the bottom line. Thus, HMM state-loops and state-transitions are represented by no-color-changes and color-changes, respectively.

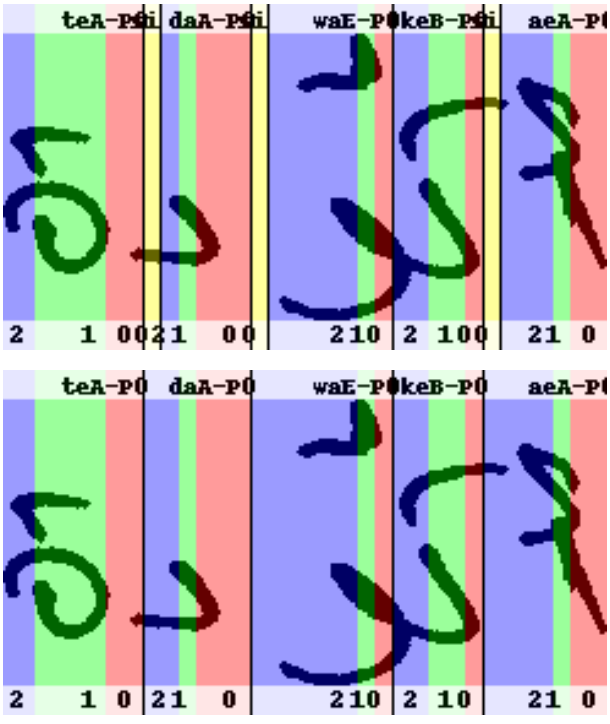
It can be observed in Figure 17 that especially the white-spaces, which can occur between compound words and pieces of Arabic words (PAW) [7], help in discriminating the isolated- (A), beginning- (B), or end-shaped (E) characters of a word w.r.t. the middle-shaped (M) characters, where usually no white-spaces occur on the left or right side of the character (cf. [32, 23] for more details about A/B/M/E shaped characters). The frames corresponding to the white-space part of the words are aligned in a more balanced way in Figure 17(a) and Figure 17(b) using the M-MMI modeling (lower rows) opposed to ML modeling (upper rows): the proposed M-MMI models learned that white-spaces help to discriminate different characters. This can even lead to a different writing variant choice without any white-space models [7] (see Figure 17(c)). Note that we cannot know in advance in training if a white-space is used or not, and if so, how large it is, as it is not transcribed in the corpora and depends on



(a)

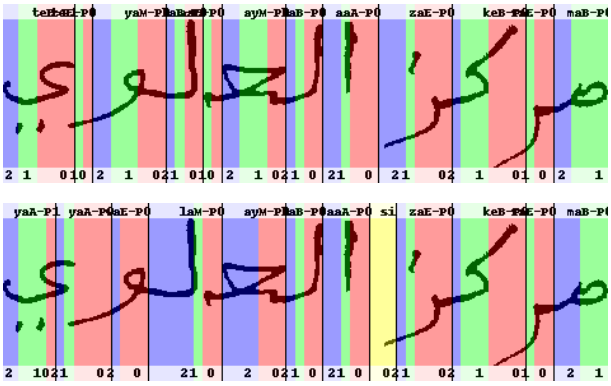


(b)

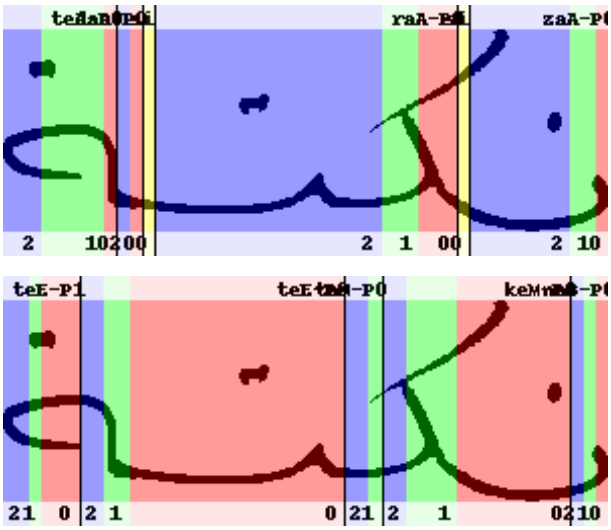


(c)

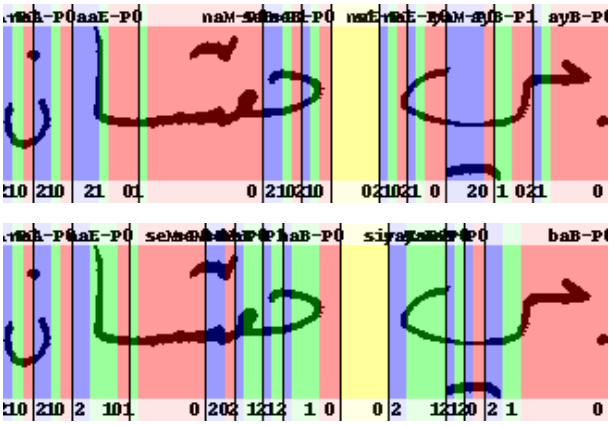
Figure 17 Supervised training alignment comparisons: The upper rows show alignments to the maximum-likelihood (ML) trained model, the lower rows to the modified maximum mutual information (M-MMI) trained models.



(a)



(b)



(c)

Figure 18 Unsupervised test alignment comparisons: The upper rows show incorrect unsupervised alignments to the maximum-likelihood (ML) trained model, the lower rows correct unsupervised alignments to the modified maximum mutual information (M-MMI) trained models.

the writer's handwriting style (e.g. cursive style used in Figure 17(a)).

In Figure 18, unsupervised test alignments are compared. The upper rows show incorrectly recognized words by unsupervised alignments to the ML trained model, the lower rows correctly recognized words by unsupervised alignments to the M-MMI trained models. Due to the discriminatively trained character models, the alignment in Figure 18(a) to the M-MMI model is clearly improved over the ML model, and the system opts for the correct compound-white-space writing variant [7]. In Figure 18(b), again the alignment is improved by the discriminatively trained white-space and character models. Figure 18(c) shows a similar alignment to the white-space model, but a clearly improved and correct alignment to the discriminatively trained character models.

Similar alignment observations can be made for the IAM database, especially for punctuation and white-space symbols.

4.4 Comparisons with other Systems

IFN/ENIT database and ICDAR Competitions. In Table 8 we compare or own evaluation results on the ICDAR 2005 [25] setups (without any tuning on test data as explained in Section 4.2) and ICDAR 2009 [24] setups. It should be noted that the result for the *abcd-e* condition is one of the best known error rates in the literature [9].

The ICDAR 2009 test datasets which are unknown to all participants were collected for the tests of the ICDAR 2007 competition. The words are from the same lexicon as those of the IFN/ENIT database and written by writers, who did not contribute to the data sets before, and are separated into set f and set s. Our results (externally calculated by TU Braunschweig) in Table 8 ranked third at the ICDAR 2009 competition and are among the best purely HMM based systems, as the A2iA and MDLSTM systems are hybrid system combinations or full neural network based systems, respectively. Also note that our single HMM based system is better than the independent A2iA systems (cf. [24] for more details), and that the results confirm that our proposed M-MMI-conf approach even generalizes well on the more difficult set s.

Note the 36% relative improvement in Table 8 we achieved in the recent ICFHR 2010 Arabic handwriting competition [26] with the proposed M-MPE training framework but an MLP based feature extraction (not described here), and again without system-combinations.

Table 9 Evaluation and comparison of the proposed confidence-based model adaptation methods on the large-vocabulary line recognition task of the IAM database: training is measured after 100 Rprop iterations, the corresponding confidence-based adaptations are measured after 15 Rprop iterations

Systems	WER [%]		CER [%]	
	Devel	Eval	Devel	Eval
RWTH OCR (this work)				
ML baseline [19]	31.9	38.9	8.4	11.8
+ M-MMI	25.8	31.6	7.6	11.8
+ M-MMI-conf	23.7	**29.0**	6.8	10.5
+ M-MPE	24.3	30.0	6.9	10.9
+ M-MPE-conf	**23.7**	29.2	**6.5**	**10.3**
Bertolami et al. [3] (HMM)	30.9	35.5	-	-
E. et al. [10] (HMM)	32.8	38.8	-	18.6
Natarajan et al. [28] (HMM)	-	*40.0**	-	-
Romero et al. [38] (HMM)	*30.6**	-	-	-
Bertolami et al. [3] (HMMs)	26.8	32.8	-	-
Graves et al. [13] (RNN)	-	25.9	-	18.2
E. et al. [10] (HMM/ANN)	19.0	22.4	-	9.8

(* different training/testing data, only qualitative comparison)

IAM Database. Summarizing results and comparisons of the proposed confidence-based model adaptation methods on the large-vocabulary line recognition task of the IAM database are reported in Table 9. It can be seen that the performance of our ML trained baseline system [19] is among current state-of-the-art systems [3, 10], and that our proposed confidence- and margin-based extensions of the discriminative MMI/MPE training criteria achieve the currently best known WERs/CERs for a purely HMM based system using a very simple feature extraction. Even ensemble based HMM approaches as proposed in [3] are outperformed by our approaches.

5 Conclusions

We presented a novel confidence- and margin-based discriminative training using a MMI/MPE training criterion for model adaptation in offline handwriting recognition. The advantages of the proposed methods using an HMM based multi-pass decoding system were shown for Arabic handwriting on the IFN/ENIT corpus (isolated word recognition) and for Latin handwriting on the IAM corpus (large-vocabulary, continuous sentence recognition). Both approaches showed their robustness w.r.t. transcription errors and outperformed the maximum-likelihood (ML) trained baseline models.

We discussed an approach how to modify existing training criteria for handwriting recognition like for example MMI and MPE to include a margin term. The modified training criterion M-MMI was shown to be

Table 8 Comparison to ICDAR 2005/2009 and ICFHR 2010 Arabic handwriting recognition competition results on the IFN/ENIT database

Competition	Group	WER [%]			
		abc-d	abcd-e	abcde-f	abcde-s
ICDAR 2005 [25]	UOB	15.0	24.1		
	ARAB-IFN	12.1	25.3	-	-
	ICRA (Microsoft)	11.1	34.3	-	-
ICDAR 2009 [24]	MDLSTM	-	-	**6.6**	18.9
	A2iA (combined)	-	-	10.6	23.3
	(NN/HMM)	-	-	14.4	29.6
	(HMM)	-	-	17.8	33.6
	RWTH OCR (this work, M-MMI)	6.1	15.4	14.5	28.7
	RWTH OCR (this work, M-MMI-conf)	6.0	14.6	14.3	27.5
	UOB-ENST (HMM, combined)	-	-	16.0	27.7
ICFHR 2010 [26]	UPV PRHLT (HMM)	7.5	12.3	7.8	**15.4**
	RWTH OCR (this work, w/ MLP features)	**3.5**	**7.3**	9.1	18.9
	UPV PRHLT (HMM, w/o vert. norm.)	-	-	12.1	21.6
	CUBS-AMA (HMM)	-	-	19.7	32.1
Other results	BBN [28]	10.5	-	-	-
	SIEMENS [39]	-	18.1	12.8	26.1

closely related to existing large margin classifiers (e.g. SVMs) with the respective loss function. This approach allows for the direct evaluation of the utility of the margin term for handwriting recognition. As expected, the benefit from the additional margin term clearly depends on the training conditions. The proposed discriminative training approach could outperform the ML trained system on all tasks.

The impact of different writing styles was dealt with a novel confidence-based discriminative training for model adaptation, where the usage of state-confidences during the iterative optimization process based on the modified M-MMI-conf criterion could decrease the word-error-rate on the IFN/ENIT database by 33% relative in comparison to a ML trained system.

On the IAM database, similar improvements could be observed for the proposed M-MMI-conf and M-MPE-conf criteria, leading to a WER decrease by 25% relative in comparison to a maximum-likelihood trained system, and representing one of the best known 29% WER in the literature for a single and purely HMM based system. In supervised training, the M-MPE criterion could outperform the M-MMI approach, whereas in unsupervised and confidence-based model adaptation, the M-MMI-conf approach could clear the initial gap to the M-MPE trained model.

Interesting for further research will remain hybrid HMM/ANN approaches [13, 10], combining the advantages of large and non-linear context modeling via neural networks while profiting from the Markovian sequence modeling. This is also supported by the 36% relative improvement we could achieve in the ICFHR 2010 Arabic handwriting competition [26] with the proposed framework but an MLP based feature extraction.

Acknowledgements. We would like to thank David Rybach and Christian Gollan for their support. This work was partly realized as part of the Quaero Programme, funded by OSEO, French State agency for innovation.

References

1. T. Anastasakos and S.V. Balakrishnan. The use of confidence measures in unsupervised adaptation of speech recognizers. In *International Conference on Spoken Language Processing*, Sydney, Australia, 1998.
2. I. Bazzi, R. Schwartz, and J. Makhoul. An omnifont open-vocabulary ocr system for english and arabic. *IEEE TPAMI*, 21(6):495–504, 1999.
3. R. Bertolami and H. Bunke. Hidden markov model-based ensemble methods for offline handwritten text line recognition. *Pattern Recognition*, 41(11):3452–3460, November 2008.
4. Roman Bertolami and Horst Bunke. Hmm-based ensamble methods for offline handwritten text line recognition. *Pattern Recognition*, 41:3452–3460, 2008.
5. Alain Biem. Minimum classification error training for online handwriting recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(7):1041–1051, 2006.
6. Philippe Dreuw, Georg Heigold, and Hermann Ney. Confidence-based discriminative training for model adaptation in offline arabic handwriting recognition. In *International Conference on Document Analysis and Recognition*, pages 596–600, Barcelona, Spain, July 2009.
7. Philippe Dreuw, Stephan Jonas, and Hermann Ney. White-space models for offline arabic handwriting recognition. In *International Conference on Pattern Recognition*, Tampa, Florida, USA, December 2008.
8. Philippe Dreuw, David Rybach, Christian Gollan, and Hermann Ney. Writer adaptive training and writing variant

model refinement for offline arabic handwriting recognition. In *ICDAR*, Barcelona, Spain, July 2009.
9. Haikal El Abed and Volker Märgner. Improvement of arabic handwriting recognition systems: combination and/or reject? In *Document Recognition and Retrieval XVI*, volume 7247 of *SPIE*, San Jose, CA, USA, January 2009.
10. S. Espana-Boquera, M. Castro-Bleda, J. Gorbe-Moya, and F. Zamora-Martinez. Improving offline handwritten text recognition with hybrid hmm/ann models. *IEEE TPAMI*, PP(99):pre–print, 2010.
11. Gernot A. Fink and Thomas Plötz. Unsupervised estimation of writing style models for improved unconstrained off-line handwriting recognition. In *International Workshop on Frontiers in Handwriting Recognition*, La Baule, France, October 2006.
12. Christian Gollan and Michiel Bacchiani. Confidence scores for acoustic model adaptation. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 4289–4292, Las Vegas, NV, USA, April 2008.
13. A. Graves, M. Liwicki, S. Fernandez, R. Bertolami, H. Bunke, and J. Schmidhuber. A novel connectionist system for unconstrained handwriting recognition. *IEEE TPAMI*, 31(5):855–868, May 2009.
14. G. Heigold, R. Schlüter, and H. Ney. On the equivalence of Gaussian HMM and Gaussian HMM-like hidden conditional random fields. In *INTERSPEECH*, Antwerp, Belgium, August 2007.
15. Georg Heigold. *A Log-Linear Discriminative Modeling Framework for Speech Recognition*. PhD thesis, RWTH Aachen University, Aachen, Germany, June 2010.
16. Georg Heigold, Thomas Deselaers, Ralf Schlüter, and Hermann Ney. Modified mmi/mpe: A direct evaluation of the margin in speech recognition. In *ICML*, pages 384–391, Helsinki, Finland, July 2008.
17. Georg Heigold, Philippe Dreuw, Stefan Hahn, Ralf Schlüter, and Hermann Ney. Margin-based discriminative training for string recognition. *Journal of Selected Topics in Signal Processing - Statistical Learning Methods for Speech and Language Processing*, 4(6):917–925, December 2010.
18. T. Jebara. *Discriminative, generative, and imitative learning*. PhD thesis, Massachusetts Institute of Technology, 2002.
19. S. Jonas. Improved modeling in handwriting recognition. Master's thesis, Human Language Technology and Pattern Recognition Group, RWTH Aachen University, Aachen, Germany, Jun 2009.
20. A. Juan, A. H. Toselli, J. Domnech, J. Gonzlez, I. Salvador, E. Vidal, and F. Casacuberta. Integrated handwriting recognition and interpretation via finite-state models. *International Journal of Pattern Recognition and Artifial Intelligence*, 2004:519–539, 2001.
21. T. Kemp and T. Schaaf. Estimating confidence using word lattices. In *European Conference on Speech Communication and Technology*, Rhodes, Greece, 1997.
22. R. Kneser and H. Ney. Improved backing-off for m-gram language modeling. In *IEEE ICASSP*, volume 1, pages 49–52, Detroit, MI, 1995.
23. Liana M. Lorigo and Venu Govindaraju. Offline Arabic handwriting recognition: A survey. *IEEE PAMI*, 28(85):712–724, May 2006.
24. V. Märgner and H. El Abed. ICDAR 2009 Arabic handwriting recognition competition. In *ICDAR*, pages 1383–1387, Barcelona, Spain, July 2009.
25. V. Märgner, M. Pechwitz, and H.E. Abed. ICDAR 2005 Arabic handwriting recognition competition. In *ICDAR*, volume 1, pages 70–74, Seoul, Korea, August 2005.
26. Volker Märgner and Haikal El Abed. ICFHR 2010 arabic handwriting recognition competition. In *ICFHR*, November 2010.
27. U.-V. Marti and H. Bunke. The iam-database: an english sentence database for offline handwriting recognition. *International Journal on Document Analysis and Recognition*, 5(1):39–46, November 2002.
28. P. Natarajan, S. Saleem, R. Prasad, E. MacRostie, and K. Subramanian. *Arabic and Chinese Handwriting Recognition*, volume 4768/2008 of *LNCS*, chapter Multi-lingual Offline Handwriting Recognition Using Hidden Markov Models: A Script-Independent Approach, pages 231–250. Springer Berlin / Heidelberg, 2008.
29. R. Nopsuwanchai and D. Povey. Discriminative training for HMM-based offline handwritten character recognition. In *ICDAR*, pages 114–118, 2003.
30. Roongroj Nopsuwanchai, Alain Biem, and William F. Clocksin. Maximization of mutual information for offline thai handwriting recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(8):1347–1351, 2006.
31. M. Padmanabhan, G. Saon, and G. Zweig. Lattice-based unsupervised mllr for speaker adaptation. In *ISCA ITRW Automatic Speech Recognition: Challenges for the Millenium*, Paris, France, 2000.
32. M. Pechwitz, S. Snoussi Maddouri, V. Mägner, N. Ellouze, and H. Amiri. IFN/ENIT-database of handwritten Arabic words. In *Colloque International Francophone sur l'Ecrit et le Document (CIFED)*, Hammamet, Tunis, October 2002.
33. M. Pitz, F. Wessel, and H. Ney. Improved mllr speaker adaptation using confidence measures for conversational speech recognition. In *International Conference on Spoken Language Processing*, Beijing, China, 2000.
34. M. Pitz, F. Wessel, and H. Ney. Improved mllr speaker adaptation using confidence measures for conversational speech recognition. In *Int. Conf. on Spoken Language Processing*, Beijing, China, 2000.
35. D. Povey. *Discriminative Training for Large Vocabulary Speech Recognition*. PhD thesis, Cambridge, England, 2004.
36. D. Povey, D. Kanevsky, B. Kingsbury, B. Ramabhadran, G. Saon, and K. Visweswariah. Boosted MMI for model and feature-space discriminative training. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Las Vegas, NV, USA, April 2008.
37. D. Povey and P. C. Woodland. Minimum phone error and I-smoothing for improved discriminative training. In *International Conference on Acoustics, Speech, and Signal Processing*, volume 1, Orlando, FL, 2002.
38. V. Romero, V. Alabau, and J. M. Benedi. Combination of n-grams and stochastic context-free grammars in an offline handwritten recognition system. *Lecture Notes in Computer Science*, 4477:467–474, 2007.
39. M.-P. Schambach, J. Rottland, and T. Alary. How to convert a latin handwriting recognition system to arabic. In *ICFHR*, 2008.
40. R. Schlüter, B. Müller, F. Wessel, and H. Ney. Interdependence of language models and discriminative training. In *IEEE Automatic Speech Recognition and Understanding Workshop*, volume 1, pages 119–122, Keystone, CO, December 1999.
41. Ralf Schlüter. *Investigations on Discriminative Training Criteria*. PhD thesis, RWTH Aachen University, Aachen, Germany, September 2000.
42. J. Zhang, R. Jin, Y. Yang, and A.G. Hauptmann. Modified logistic regression: An approximation to SVM and its applications in large-scale text categorization. In *ICML*, August 2003.