

# 1 Joint semi-supervised Learning of Hidden Conditional 2 Random Fields and Hidden Markov Models

3 Yann Soullard, Martin Saveski, Thierry Artières

4 *LIP6*  
5 *Université Pierre et Marie Curie*  
6 *4 place Jussieu, 75005, Paris, France*  
7 *firstname.lastname@lip6.fr*

---

## 8 **Abstract**

9 Although semi-supervised learning has generated great interest for designing  
10 classifiers on vector data there has been comparatively very few works on  
11 semi-supervised learning for structured outputs like sequences. We investi-  
12 gate here semi-supervised approaches for learning hidden state conditional  
13 random fields that have been proposed recently for sequence and signals clas-  
14 sification. In particular we propose a new approach to deal with this problem  
15 that relies on an iterative joint learning of a pair of discriminative-generative  
16 models, namely Hidden Markov Models (HMMs) and Hidden Conditional  
17 Random Fields (HCRFs). The method builds on rather simple strategies for  
18 semi-supervised learning of HMMs and on strategies for learning a HCRF  
19 from a HMM. This algorithm has connections with few previous works such  
20 as co-training. We investigate the behavior of the method on artificial data  
21 and provide experimental results on two real datasets for handwriting letters  
22 classification and chart pattern recognition. All along the experimental sec-  
23 tion, we compare our approach with state of the art semi-supervised methods,  
24 entropy minimization and co-training.

25 *Keywords:* Hidden Markov Models, Hidden Conditional Random Fields,  
26 semi-supervised learning, co-training

---

## 27 **1. Introduction**

28     Sequence classification and sequence labeling are fundamental tasks oc-  
29 ccurring in many application domains such as speech, financial time series, and  
30 handwriting. The most popular method for such tasks are the well known  
31 Hidden Markov Models (HMMs) [1]. HMMs are generative models which are  
32 trained to maximize the joint likelihood of observation sequences and of their  
33 labeling. HMMs benefit from efficient algorithms both for inference and for  
34 training but suffer few severe drawbacks. In particular they are traditionally  
35 learned via maximum likelihood estimation, which is a non discriminative  
36 training criterion.

37     Although some attempts have been made to overcome this limitation by  
38 learning discriminatively HMM systems through the optimization of a dis-  
39 criminant criterion like minimum error classification [2], perceptron loss [3],  
40 maximum mutual information [4], margin maximization [5, 6]. A more  
41 straightforward way to reach higher discriminative power is to define a model  
42 of the posterior conditional probability (i.e. the probability of the label given  
43 the input sequence). Hidden Conditional Random Fields (HCRFs) are such  
44 models. They are a variant of Conditional Random Fields (CRFs) [7] that  
45 make use of hidden states to account for the underlying structure of the data  
46 (alike HMMs). They have been used for various signal labeling tasks, in par-  
47 ticular for speech signals [8], [9], eye movements [10], handwriting [11, 12],  
48 gestures and images [13], financial time series [14].

49 When building a classification system, whatever the model one chooses  
 50 to design a sequence labeling system one has first to gather, then to label,  
 51 a sufficiently large training corpus. This always comes with a cost that may  
 52 make problematic the design of a good system. This motivated the study  
 53 of semi-supervised learning (SSL), which has been proposed first for vector  
 54 data. SSL aims at learning classifiers based on both labeled samples (usually  
 55 few) and unlabeled samples (usually many). A number of SSL methods have  
 56 been proposed, like entropy based methods [15], margin based methods [16],  
 57 co-training algorithms [17] (see [18] for reviews).

58 Few works have investigated semi-supervised learning for sequential and  
 59 more generally structured data. Some of these investigated HMMs semi-  
 60 supervised learning for speech recognition and text classification tasks [19],  
 61 [20],[21], but the conclusion of these works are rather mitigated, where SSL  
 62 has been shown to eventually degrade performance with respect to purely  
 63 supervised training [22], [23]. It appears in the literature that SSL may be  
 64 less efficient for learning more complex models, such as HMMs which include  
 65 hidden states to deal with partially observed data. Besides, some other works  
 66 attempted to learn CRFs in a semi-supervised setting for language processing  
 67 and biological problems, yielding some significant improvements [24], [25]. It  
 68 is worth noticing that few of these works rely on designing a hybrid model  
 69 mixing HMMs and CRFs where HMMs are learned in a semi-supervised way,  
 70 making indirectly the learning of CRFs to be semi-supervised [25]. At the  
 71 end we are not aware of any work today on SSL algorithms for complex  
 72 discriminative models such as HCRFs excepted in our previous works [26].

73 We propose in this study to investigate the relevance of previous SSL

74 schemas to the learning of HCRF and we propose a new algorithm for semi-  
 75 supervised learning of HCRFs. It relies on an iterative joint learning of a  
 76 pair of generative and discriminative models, namely HMMs and HCRFs. It  
 77 relies on the relative easiness of SSL for HMMs and on recent results showing  
 78 how one can initialize a HCRF from a HMM system [8]. In our approach,  
 79 HMMs learning makes explicit use of both labeled and unlabeled data while  
 80 HCRF learning is purely supervised and indirectly exploits unlabeled data  
 81 through its initialization from the HMM system. All along the paper we  
 82 focus on sequence classification where one wants to assign a single label to  
 83 an input sequence but extension to sequence labeling is in our mind and our  
 84 next step.

85 We first recall basics of HMMs and HCRFs in section 2 and we discuss  
 86 related works on semi-supervised learning in section 3. Then, we detail our  
 87 approach in section 4. We report experimental results on artificial data  
 88 in section 5 and investigate the behavior of our approach on handwriting  
 89 recognition and on financial chart pattern classification in section 6.

## 90 **2. Markovian models for sequence classification**

91 In this study, we focus on sequence classification where training samples  
 92 are couples  $(\mathbf{x}, y)$ , where we note  $\mathbf{x} = (\mathbf{x}_1, \dots, \mathbf{x}_T) \in \mathcal{X}$  an input sequence of  
 93 length  $T$  (sequences are noted in bold), with each frame  $\mathbf{x}_t \in \mathbb{R}^d$  being a  
 94  $d$ -dimensional feature vector which characterizes locally the input sequence,  
 95 and  $y \in \mathcal{Y}$  is its class (i.e. label).

96 *2.1. From Hidden Markov Models to Hidden Conditional Random Fields*

97 We briefly recall basics of Hidden Markov Models (HMMs) then we de-  
 98 tail Hidden Conditional Random Fields (HCRFs). Note that we focus on  
 99 sequence classification all along the presentation.

100 HMM are generative models which define a joint probability over an ob-  
 101 servation sequence and its class  $p(\mathbf{x}, y)$ . To account for variability of obser-  
 102 vations HMMs include hidden states which are not observed. Hence training  
 103 data are incomplete in that the state sequence corresponding to a particular  
 104 observation sequence is unknown. As a result the joint probability of an  
 105 observed sequence and of its class is the sum over all hidden state sequences  
 106 of the joint probability of the observed sequence and of the hidden state  
 107 sequence:

$$p(\mathbf{x}, y|\Theta) = \sum_{\mathbf{h}} p(\mathbf{x}, \mathbf{h}|y, \Theta) p(y|\Theta) \quad (1)$$

108 where  $\Theta$  stands for the HMM parameters  $\mathbf{h}$  denotes a hidden state sequence,  
 109 with  $\forall t, h_t \in S$  with  $S = \{s_1, \dots, s_Q\}$  being the set of the  $Q$  states of the  
 110 model. In the formula above the summation over  $\mathbf{h}$  is taken over all  $\mathbf{h}$  that  
 111 match with the labeling  $y$ , that we will note  $S(y)$  in the following.

112 The joint probability  $p(\mathbf{x}, \mathbf{h}|y)$  may be factorized according to (taking the  
 113 logarithm):

$$\log p(\mathbf{x}, \mathbf{h}|y) = \log p(h_1) + \sum_{t=2}^T \log p(h_t|h_{t-1}) + \sum_{t=1}^T \log p(\mathbf{x}_t|h_t) \quad (2)$$

114 where, using HMM's standard terminology,  $p(h_1)$  are initial probabilities,  
 115  $p(h_t|h_{t-1})$  are transition probabilities, and  $p(\mathbf{x}_t|h_t)$  are emission probabilities.

116 HMMs are usually learned in a non discriminative way to maximize the  
 117 likelihood of training data. Traditionally for dealing with signal like data  
 118 (speech, handwriting, etc), one uses a left-right HMM topology for every  
 119 class, where transitions are allowed from any state to itself or to the next  
 120 state.

121 Hidden CRFs (HCRFs) have been proposed as an extension of CRFs (that  
 122 were initially proposed in [7] for text data) for dealing with more complex  
 123 and structured data [27] [11] [8]. In CRF-based systems there is usually one  
 124 state per class (e.g. a POS tag) while there are several states corresponding  
 125 to a given class in HCRF, alike in HMMs. Given a HCRF with parameters  
 126  $\Lambda$  the class conditional probability of a class  $y$  given an input sequence  $\mathbf{x}$   
 127 defined as:

$$p(y|\mathbf{x}, \Lambda) = \frac{1}{Z(\mathbf{x}, \Lambda)} \sum_{\mathbf{h} \in S(y)} \exp^{<\Lambda, \Phi(\mathbf{x}, y, \mathbf{h})>} \quad (3)$$

128 where  $\Phi(\mathbf{x}, y, \mathbf{h})$  is a joint feature vector corresponding to a state sequence  $\mathbf{h}$ ,  
 129 and  $Z(\mathbf{x}, \Lambda) = \sum_{y'} \sum_{\mathbf{h}' \in S(y')} \exp^{<\Lambda, \Phi(\mathbf{x}, y', \mathbf{h}')>}$  is a normalization term. When  
 130 given an input sequence  $\mathbf{x}$ , its predicted class is determined according to  
 131  $\text{argmax}_y p(y|\mathbf{x}, \Lambda)$ .

132 To make the model tractable one assumes the feature vector  $\Phi$  to be de-  
 133 composable. Usually this decomposition relies on a Markov network encoding  
 134 conditional dependencies between random variables to infer ( $y$  and  $\mathbf{h}$ ). In  
 135 sequence modeling one often considers a decomposition over time steps :

$$\Phi(\mathbf{x}, y, \mathbf{h}) = \sum_t \phi(\mathbf{x}, y, \mathbf{h}, t) \quad (4)$$

136 where  $\phi(\mathbf{x}, y, \mathbf{h}, t)$  is a *local* feature function. More precisely on exploits  
 137 a Markov network with two types of cliques: transition cliques involving  
 138 two successive states and local cliques involving a state at a particular time  
 139 and the observation at that time. With these assumptions, the conditional  
 140 probability above is usually rewritten as:

$$p(y|\mathbf{x}, \Lambda) = \frac{1}{Z(\mathbf{x}, \Lambda)} \sum_{\mathbf{h} \in S(y)} \exp^{\sum_t \langle \boldsymbol{\lambda}^{trans}, \boldsymbol{\phi}^{trans}(\mathbf{x}, y, h_t, h_{t-1}) \rangle + \langle \boldsymbol{\lambda}^{loc}, \boldsymbol{\phi}^{loc}(\mathbf{x}, y, h_t) \rangle} \quad (5)$$

141 where  $\boldsymbol{\lambda}^{loc}$  is the subset of  $\Lambda$  weighting the features for each state,  $\boldsymbol{\lambda}^{trans}$  is  
 142 the subset of  $\Lambda$  weighting the transition features between states,  $\boldsymbol{\phi}^{loc}$  and  
 143  $\boldsymbol{\phi}^{trans}$  are joint feature vectors for transition and local cliques.

## 144 2.2. Learning

145 HMMs are traditionally learned via the Baum-Welsh algorithm [28] to  
 146 maximize the likelihood of the training data:

$$147 \quad \mathcal{L}(\Theta) = \sum_{i=1}^n \log(p(y^i, \mathbf{x}^i | \Theta)) \quad (6)$$

148 where  $\Theta$  denotes the parameter set of the HMMs of all classes, and the su-  
 149 perscript  $i$  refers to the number of the training sample, i.e.  $\mathbf{x}^i$  stands for the  
 150  $i^{th}$  training sequence and  $y^i$  for its class.

151

152 Alike CRFs, HCRFs are usually learned to maximize the conditional like-  
 153 lihood:

$$\mathcal{L}(\Lambda) = \sum_{i=1}^n \log(p(y^i | \mathbf{x}^i, \Lambda)) \quad (7)$$

154 The optimization is traditionally performed with gradient like algorithms,  
 155 stochastic gradient [29], or second order extensions like LBFGS [30].

156 One limit of HCRFs lies in the non convexity of the training criterion,  
 157 which comes from the introduction of hidden states just like for HMMs. This  
 158 makes training sensitive to initialization and easily lead to overfitting. To  
 159 overcome this problem few solutions have been proposed to initialize HCRFs.  
 160 The most interesting one has been proposed in [8], it consists in learning first  
 161 a HMM system, then to initialize the HCRF parameters so that it reproduces  
 162 the classification behavior of the HMMs (note that the HCRF topology must  
 163 match the HMM one). We briefly explain how this can be done.

164 The key point is that the joint log likelihood of an input sequence and of a  
 165 sequence of states, as computed by a HMM, may be written as a dot product  
 166 between a particular parameter vector and a joint feature map depending  
 167 on the class, the sequence of hidden states and the input sequence. Indeed,  
 168 noting  $\Theta$  the parameter set of an HMM, for any state sequence  $\mathbf{h}$  we have:

$$\begin{aligned} \log p(\mathbf{x}, y, \mathbf{h} | \Theta) &= \log(\pi_{h_1}) + \log(p(\mathbf{x}_1 | h_1)) \\ &+ \sum_{t=2}^T \left( \log p(h_t | h_{t-1}) + \log p(\mathbf{x}_t | h_t, \boldsymbol{\mu}_{h_t}, \Sigma_{h_t}) \right) \end{aligned}$$

169

170

with, following [31]:

$$\begin{aligned} \log p(\mathbf{x}_t | h_t, \Theta) &= \frac{1}{2} \left( \mathbf{x}_t^T \Sigma_{h_t}^{-1} \mathbf{x}_t - \mathbf{x}_t^T \Sigma_{h_t}^{-1} \boldsymbol{\mu}_{h_t} - \boldsymbol{\mu}_{h_t}^T \Sigma_{h_t}^{-1} \mathbf{x}_t \right. \\ &\quad \left. + \boldsymbol{\mu}_{h_t}^T \Sigma_{h_t}^{-1} \boldsymbol{\mu}_{h_t} - \log((2\pi)^d |\Sigma_{h_t}|) \right) \end{aligned}$$



171 Hence, we may write:

$$\log p(\mathbf{x}, y, \mathbf{h} | \Theta) = \langle \boldsymbol{\lambda}, \boldsymbol{\phi}(\mathbf{x}, y, \mathbf{h}) \rangle$$

172

173 which may be rewritten, using appropriate definitions (see Appendix A for  
174 details), as:

$$\langle \boldsymbol{\lambda}, \boldsymbol{\phi}(\mathbf{x}, y, \mathbf{h}) \rangle = \sum_t \langle \boldsymbol{\lambda}^{trans}, \boldsymbol{\phi}^{trans}(\mathbf{x}, y, h_t, h_{t-1}) \rangle + \langle \boldsymbol{\lambda}^{loc}, \boldsymbol{\phi}^{loc}(\mathbf{x}, y, h_t) \rangle$$

175 The above result yields an efficient learning procedure for learning HCRFs  
176 for sequence classification. First, one learns a HMM system with one HMM  
177 per class. Then one initializes a HCRF system with the same topology with  
178 the above formulas. This HCRF system outputs exactly the same classifica-  
179 tion decision as the HMM system. Finally one uses the standard discrimina-  
180 tive conditional likelihood criterion of HCRFs to fine tune the HCRF system.  
181 At the end, the initialization by the HMM system allows starting the HCRF  
182 optimization process in an interesting area so as to reach a relevant local  
183 minimum of the non convex HCRF optimization criterion.

### 184 3. State of the art in semi-supervised learning

185 Although semi-supervised learning has received great attention in the  
186 last ten years there has been only few attempts for dealing with structured  
187 data and complex models like CRFs and HCRFs. Most works concern vector  
188 data, with some exceptions like [32]. We first review main categories of semi-  
189 supervised learning methods then we present in more details works that have  
190 focused on SSL for sequential data and in particular for markovian models.

### 191 3.1. *Generic methods*

192 We provide here a brief overview of the main categories of methods, a  
193 detailed survey may be found in [33]. Some of these methods are dedicated  
194 to generative models, some on discriminative models and finally few rely on  
195 a mix between generative and discriminative models.

196 *Mixture* methods relying on the learning of a mixture of generative models  
197 through Expectation Maximization have shown to improve performance by  
198 using unlabeled data, in the cases when the classes consist of well clustered  
199 data. For instance [19] apply the EM algorithm on mixture of multinomial  
200 distributions for the task of text classification. They showed that the result-  
201 ing classifiers perform better than those trained only on labeled data. Besides  
202 [34] used the same algorithm on a face orientation discrimination task.

203 *Co-training* has been popularized by [17]. It assumes that the features  
204 (samples are feature vectors) can be split in two sets (views), each of which  
205 is sufficient to train a good classifier and they are conditionally independent  
206 of one another given the class. Initially, two separate classifiers are trained  
207 on the labeled data with each feature set. Then, each classifier classifies the  
208 unlabeled data and “teaches” the other classifier with the unlabeled examples  
209 it feels most confident about. Each classifier is re-trained with the new labels  
210 and the process is repeated. In real tasks, having two sufficient and redundant  
211 views as the standard framework [17] is uncommon. [35] extend this approach  
212 to two learners which are not necessary trained on two different views and  
213 prove that this may enhance the combination of the two models against  
214 individual systems. Thus, they show that the co-training process may work  
215 well with two classifiers working on the same view of the data provided the

models are enough different. Also, the co-training offers more guarantee to work well if the two classifiers are of comparable accuracy.

*Graph-based* semi-supervised methods define a graph where the nodes are labeled and unlabeled examples in the dataset, and edges reflect similarity of the examples. Different algorithms operating on the graph have been proposed, including [36, 37]. These methods achieve good performance when similar instances in the data set have similar labels.

*Self-training* is another commonly used technique. A classifier is first trained with the small amount of labeled data and then it is used to classify the unlabeled data. Then, the most confident unlabeled points, with their predicted labels, are added to the training set. The classifier is re-trained and the procedure is repeated. This method has been successfully applied to several natural language processing tasks in [38, 39].

*Entropy Minimization Method* has been introduced in [15] by Grandvalet and Bengio. They use the label entropy on unlabeled data as a regularizer. By minimizing the entropy, the method assumes a prior which prefers minimal class overlap. For a training dataset of  $L$  labeled data  $\{(\mathbf{x}^1, y^1), \dots, (\mathbf{x}^L, y^L)\}$  and of  $U$  unlabeled data  $\{\mathbf{x}^{L+1}, \dots, \mathbf{x}^{L+U}\}$ , they propose maximizing the following objective function:

$$\mathcal{L}(\Theta) = \sum_{i=1}^L \log p(y^i | \mathbf{x}^i, \Theta) + \gamma \sum_{j=L+1}^{L+U} \sum_{y' \in \mathcal{Y}} p(y' | \mathbf{x}^j, \Theta) \log p(y' | \mathbf{x}^j, \Theta) \quad (8)$$

where,  $\gamma$  is a hyper parameter of the algorithm controlling the influence of the unlabeled data.

Finally *hybrid methods* have been proposed that mix generative and discriminative models [40] and [41]. [41] proposed first a convex combination of

the objective functions of a generative and of a discriminative model while [40] proposed to optimize the following objective function :

$$\mathcal{L}(\Theta, \Lambda) = \sum_{i=1}^L \log p(y^i | \mathbf{x}^i, \Lambda) + \sum_{j=1}^{L+U} \log p(\mathbf{x}^j | \Theta) + \log(p(\Theta, \Lambda)) \quad (9)$$

where the prior  $p(\Theta, \Lambda)$  allows blending generative and discriminative approaches. In the particular case where the prior is uniform the generative and discriminative models are decoupled so that the discriminative model is learned in a fully supervised setting. In the case where the prior is peaked on  $\Theta = \Lambda$  the two models are constrained so that one recovers the approach in [41]. Finally if the prior is smoother, the discriminative model is learned in a supervised setting with the constraint of being not too far from the generative model, which is learned in a semi-supervised setting.

### 3.2. Semi-supervised learning of markovian models for sequence classification

A number of works have proposed methods for semi-supervised learning of HMMs and of CRFs, they belong to above categories.

*Mixture approach.* The Mixture approach has been applied to HMMs in [19], [20]. In this setting, the likelihood-based criterion is defined as:

$$\mathcal{L}(\Theta) = \frac{(1-\gamma)}{L} \sum_{i=1}^L \log p(\mathbf{x}^i, y^i | \Theta) + \frac{\gamma}{U} \sum_{j=L+1}^{L+U} \log p(\mathbf{x}^j | \Theta) \quad (10)$$

where the parameter  $\gamma \in [0, 1]$  allows tuning the respective influence of labeled and of unlabeled data. The fully supervised and the fully unsupervised

cases are special cases when  $\gamma$  is respectively set to 0 and to 1 [42]. This approach has been observed to eventually degrade the performances for learning HMMs [22, 23].

*Entropy minimization.* [32] devises a semi-supervised variant of the support vector machine based on a co-training algorithm. The main approach extends the minimum entropy regularization framework [42] to CRFs with a regularized objective function that combines unlabeled conditional entropy and labeled conditional likelihood [24] :

$$\mathcal{L}_\gamma(\Lambda) = -\frac{\|\Lambda\|^2}{2} + \sum_{i=1}^L \log p(y^i|\mathbf{x}^i, \Lambda) + \gamma \sum_{j=L+1}^{L+U} \sum_{y \in \mathcal{Y}} p(y|\mathbf{x}^j, \Lambda) \log p(y|\mathbf{x}^j, \Lambda) \quad (11)$$

Similarly, [43], relying on information theory, defines the objective function as the combination of the conditional likelihood on labeled data and of the mutual information on unlabeled data .

*Hybrid approach.* few works have investigated combining a discriminative model trained on labeled data and a generative model trained with additional unlabeled samples. [44] suggested to discriminatively train a blending of I-units of discriminative models (CRFs) trained on labeled data and J-units of generative models (HMMs) whose learning exploits additional unlabeled data. In the same line of work, [25] proposed a weighted CRF where weights are marginal probabilities of input sequences obtained by a generative model trained in a semi-supervised manner :

$$\mathcal{L}(\Lambda) = \sum_{i=1}^L q(\mathbf{x}^i) \frac{1}{N_{\mathbf{x}^i}} \log p(y^i|\mathbf{x}^i, \Lambda) \quad (12)$$

278 where  $N_{\mathbf{x}}$  is the number of times  $\mathbf{x}$  has been observed in the training set (data  
279 are discrete) and  $q(\mathbf{x})$  is the marginal probability of observations which come  
280 from the generative model.

281 *Co-training.* Co-training has also been investigated with some success for  
282 generative markovian models. In particular, [45] applied the standard co-  
283 training algorithm with HMMs models for singing voice detection and [46]  
284 experimented co-training on HMMs and neural networks for handwriting  
285 recognition.

286

## 287 4. Joint semi-supervised learning for HMMs and HCRFs

288 Designing semi-supervised learning algorithms for HCRF has not really  
289 been studied up to now. A first solution is to extend traditional SSL ap-  
290 proaches to HCRFs. In particular we implemented methods that have been  
291 proposed for CRFs (i.e HCRFs without hidden states) and we will investigate  
292 the behavior of the methods in the experiments section. Here we describe a  
293 new approach where we jointly learn iteratively a HMM and a HCRF system.  
294 We first present the motivation of this work then we present in details the  
295 method.

### 296 4.1. Motivation

297 A starting point of our approach lies in general observations concern-  
298 ing training and generalization ability of generative and discriminative ap-  
299 proaches with small training datasets.

300        On the one hand, generative approaches (e.g. HMMs) rely on the learning  
301 of one model per class and build a distribution over observations. These ap-  
302 proaches may exhibit a higher bias and a lower variance than discriminative  
303 models [41]. On the other hand, the discriminative approach (e.g. HCRFs)  
304 directly model the conditional probability distribution which is more related  
305 to the classification goal. When the training set size increases towards in-  
306 finity, these approach usually exhibits better asymptotic performance (i.e.  
307 accuracy) than generative models but the convergence to their optimal be-  
308 havior may be slower. It may then happen that generative models reach their  
309 asymptotic performance faster than discriminative ones, i.e. with a smaller  
310 training set size [47]. From these general comments it may happen that gen-  
311 erative models may be more accurate with a small training dataset while  
312 discriminative models are more accurate when the training set size increases.  
313 Hence, mixing the two to get the best of the two worlds in any situation  
314 makes sense and is definitely appealing.

315        Besides, it is worth noticing that generative models allow simple semi-  
316 supervised learning through the use of mixture models and of an EM learning  
317 scheme [19] while semi-supervised training in discriminative models is less  
318 straightforward, which is particularly true for markovian models with hidden  
319 states as we discussed in previous section.

320        Our goal is to take advantage of the strenghts of each approach. Since  
321 generative models may be easily learned in a semi-supervised setting and  
322 since they may be less sensitive to overfitting, a first reasonable idea is to  
323 learn first a HMM system in SSL setting, then to learn a discriminative HCRF  
324 system initialized from the HMM system using the strategy described in

section 2.2 and Appendix A. To prevent overfitting of the HCRF system, one can train it with a regularized likelihood criterion where the regularization term constrain the solution to stay close from the HMM solution.

Going further, one may consider that the HCRF system is more powerful from the HMM it was initialized from so that one can use it to label the unlabeled data for HMM retraining. And again, one can learn a new HCRF system from the new HMM system, etc. Hence our approach bares some similarity with co-training but borrows some idea to [48] as well.

#### 4.2. Iterative Hybrid Algorithm (IHA)

Our learning method, which we call Iterative Hybrid Algorithm (IHA), is an iterative algorithm that blend generative and discriminative models in a semi-supervised framework. The main idea is to use the generative model to incorporate the additional information brought by the unlabeled data ( $U$ ), and to use the discriminative model to achieve better classification accuracy. The overall algorithm is illustrated in Figure 1. Initially, we train a generative model on labeled and unlabeled data ( $L \cup U$ ). In the main loop of the algorithm, we train a discriminative model on  $L$ , constraining its parameters to be close to the ones of the generative model it is initialized from. Then, we use the discriminative model to label part (or all) of  $U$  which we use with  $L$  to retrain a generative model in a supervised mode. We repeat this process a number of iterations or until convergence.

More formally, the method can be described as follows. As before the parameters of the discriminative models are denoted as  $\Lambda$ , and the parameters of the generative models are denoted as  $\Theta$ .



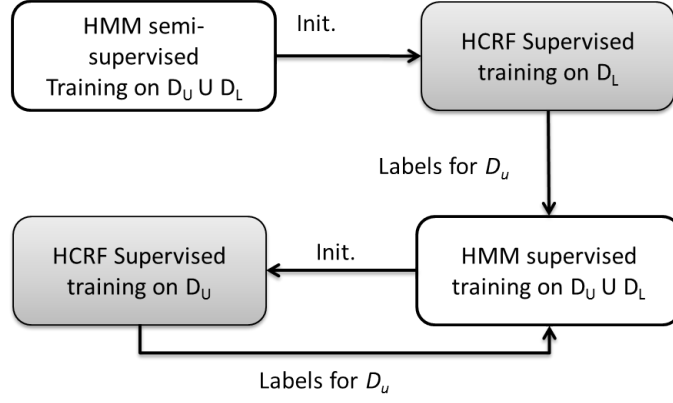


Figure 1: Semi-supervised strategy embedding HMM and HCRF learning,  $D_L$  and  $D_U$  denote the sets of labeled and unlabeled training sequences.

- 349 1. Semi-supervised learning of  $\Theta$  on  $L \cup U$  yielding  $\Theta^{(1)}$ :

$$\Theta^{(1)} = \underset{\Theta}{\operatorname{argmax}} \left( \frac{\gamma}{L} \sum_{i=1}^L \log p(\mathbf{x}^i, y^i | \Theta) + \frac{(1-\gamma)}{U} \sum_{j=L+1}^{L+U} \log \sum_{y' \in \mathcal{Y}} p(\mathbf{x}^j, y' | \Theta) \right) \quad (13)$$

- 350 2. Loop for a fixed number of iterations or until convergence. At iteration  
351  $k$ :

- 352 2.1. Supervised Learning of  $\Lambda$  on  $L$  yielding  $\Lambda^{(k)}$ , starting from  $\Theta^{(k-1)}$ :

$$\Lambda^{(k)} = \underset{\Lambda}{\operatorname{argmax}} \sum_{i=1}^L \log p(y^i | \mathbf{x}^i, \Lambda) - \frac{1}{2} \|\Lambda - \Theta^{(k-1)}\|^2 \quad (14)$$

- 353 2.2. Use  $\Lambda^{(k)}$  to label part of  $U$  which becomes  $U_{\text{Labeled}}$ , where the  
354 labels are assigned as:

$$\forall i \in [L+1, L+U], \hat{y}^i = \underset{y \in \mathcal{Y}}{\operatorname{argmax}} p(y | \mathbf{x}^i, \Lambda^{(k)}) \quad (15)$$

2.3. Supervised Learning of  $\Theta$  on  $L + U_{Labeled}$  yielding  $\Theta^{(k)}$ :

$$\begin{aligned} \Theta^{(k)} = \operatorname{argmax}_{\Theta} & \left( \frac{\gamma}{L} \sum_{i=1}^L \log p(\mathbf{x}^i, y^i | \Theta) \right. \\ & \left. + \frac{(1-\gamma)}{|U|} \sum_{j=L+1}^{L+U_{Labeled}} p(\hat{y}^j | \mathbf{x}^j, \Theta) \log p(\mathbf{x}^j | \Theta) \right) \end{aligned} \quad (16)$$

355 The only hyper parameter in the model is  $\gamma$ , which controls the influence  
 356 of the unlabeled data. Larger values of  $\gamma$  imply more influence of  $U$ . The  
 357 final output of the method are the parameters of the discriminative models  
 358  $\Lambda$  (and the parameters of the generative models  $\Theta$ ) which are used to predict  
 359 labels for new samples.

#### 360 4.3. *Alternative strategies*

361 We investigated four variants of our method that differ by the way the  
 362 generative system is retrained in step 2.3 of the algorithm described in pre-  
 363 vious section. To simplify notations, we note  $U = U_{Labeled}$  and we define, for  
 364 any class  $c \in \mathcal{Y}$ ,  $L_c$  to be the number of labeled data in class  $c$ .

365 First we may use every unlabeled sample to retrain every class model. To  
 366 do that an intuitive idea is to weight the contribution of an unlabeled sample  
 367 by its posterior probabilities as given by the discriminative system. In this  
 368 case, our algorithm comes close to the standard semi-supervised framework  
 369 of generative models. The objective for learning the whole HMM system  
 370 writes:

$$\begin{aligned}
\mathcal{L}(\Theta) = & \gamma \sum_{c \in \mathcal{Y}} \frac{1}{L_c} \sum_{i=1}^L \delta_{y^i=c} \log p(\mathbf{x}^i, y^i | \Theta) \\
& + (1 - \gamma) \frac{1}{U} \sum_{j=L+1}^{L+U} \sum_{c \in \mathcal{Y}} p(y = c | \mathbf{x}^j, \Lambda) \log p(\mathbf{x}^j, c | \Theta) \quad (17)
\end{aligned}$$

371 where  $\delta_{y=c}$  is the Dirac measure equal to 1 if  $y = c$  and else 0. We call this  
 372 variant the *AllClasses* variant since every unlabeled sample contributes to  
 373 reestimation of the HMMs of all classes.

374 A smoother version consists to retrain the HMM of a class with its corre-  
 375 sponding labeled samples and with unlabeled samples that would be affected  
 376 to this class by the HCRF system. In this case the objective for learning the  
 377 whole HMM system writes:

$$\begin{aligned}
\mathcal{L}(\Theta) = & \gamma \sum_{c \in \mathcal{Y}} \frac{1}{L_c} \sum_{i=1}^L \delta_{y^i=c} \log p(\mathbf{x}^i, y^i | \Theta) \\
& + (1 - \gamma) \sum_{c \in \mathcal{Y}} \frac{1}{\sum_{j=L+1}^{L+U} \delta_{c=\hat{y}^j}} \sum_{j=L+1}^{L+U} \delta_{c=\hat{y}^j} p(c | \mathbf{x}^j, \Theta) \log p(\mathbf{x}^j, c | \Theta) \quad (18)
\end{aligned}$$

378 where  $\hat{y}^j$  is defined as in Equation 15. We call this variant the *MaxProb*  
 379 variant since a unlabeled sample will be used to reestimate model of the  
 380 most likely class only.

Alternatively, instead of weighting the contribution of unlabeled samples  
 by their posterior probability we may simply add samples to the training  
 data set of the HMM of a class according to the HCRF decision. In this case

the objective for learning the whole HMM system writes:

$$\begin{aligned}\mathcal{L}(\Theta) = & \gamma \sum_{c \in \mathcal{Y}} \frac{1}{L_c} \sum_{i=1}^L \delta_{y^i=c} \log p(\mathbf{x}^i, y^i | \Theta) \\ & + (1 - \gamma) \sum_{c \in \mathcal{Y}} \frac{1}{\sum_{j=L+1}^{L+U} \delta_{c=\hat{y}^j}} \sum_{j=L+1}^{L+U} \delta_{c=\hat{y}^j} \log p(\mathbf{x}^j, c | \Theta)\end{aligned}\quad (19)$$

381 We call this strategy the *WeightOne* variant.

Finally, in a variant of the *WeightOne* case, one may keep only very likely samples whose conditional probability given by the discriminative model is over a threshold  $\tau$  (e.g. close to one). In such a case, we are close to a co-training like strategy where a limited number of training samples, labeled by the discriminative model, would be added to the training set of the generative one :

$$\begin{aligned}\mathcal{L}(\Theta) = & \gamma \sum_{c \in \mathcal{Y}} \frac{1}{L_c} \sum_{i=1}^L \delta_{y^i=c} \log p(\mathbf{x}^i, y^i | \Theta) \\ & + (1 - \gamma) \sum_{c \in \mathcal{Y}} \frac{1}{\sum_{j=L+1}^{L+U} \delta_{c=\hat{y}^j} \rho_\tau(\hat{y}^j, \mathbf{x}^j)} \sum_{j=L+1}^{L+U} \delta_{c=\hat{y}^j} \rho_\tau(\hat{y}^j, \mathbf{x}^j) \log p(\mathbf{x}^j, c | \Theta)\end{aligned}\quad (20)$$

382 where  $\rho_\tau(y, \mathbf{x})$  is equal to 1 if  $p(y|\mathbf{x}) \geq \tau$  otherwise it is 0. We call this  
383 strategy *SelectProb* since it relies on the selection of the likeliest samples.

384

#### 385 4.4. Discussion

386 Our method borrow ideas from co-training and from the work from [40],  
387 [48]. Indeed the way the learned HCRF influences the HMM learning is  
388 close to the general co-training idea where one model labels samples that

389 are added to the training set of another classifier. And co-training has been  
 390 proved efficient in many situations [17] [35] [45]. Moreover, one may note  
 391 that if the number  $U$  of unlabeled data predicted by the discriminative model  
 392 is much larger than the number of labeled data  $L$ , then labeled data may  
 393 become negligible so that the training set of the generative model follows the  
 394 distribution given by the discriminative model. Thus, the generative model  
 395 will tend to be as close as possible to the discriminative one, which we may  
 396 expect is better than the generative model at previous iteration.  
 397 Indeed, the second idea of the algorithm is to learn the discriminative model  
 398 in a purely supervised way on the labeled training dataset by starting from  
 399 the HMM solution and regularizing around this initial solution. We can then  
 400 expect that the HCRF solution will be a local optimum of the conditional  
 401 likelihood criterion which is close to the HMM system. Being not far from  
 402 the HMM solution one can expect the HCRF solution to indirectly take into  
 403 account the unlabeled data. And if the regularization is strong enough one  
 404 can expect that the HCRF solution will be better than the initial solution,  
 405 i.e. the HMM system, with respect to the conditional likelihood criterion.  
 406 Then there are some reason to think that the HCRF solution will be better  
 407 than the HMM solution.

## 408 **5. Experiments on artificial data sets**

409 In this section, we illustrate on synthetic data performances of our Itera-  
 410 tive Hybrid Algorithm (IHA) compared to benchmark methods : the Entropy  
 411 Minimization [15] (EM) and the Hybrid Model [40], [48] (HM).

412 Data are chosen to be as simple as possible in order to enable visual

413 investigation. We focus on a binary classification where data are generated by  
 414 two Gaussian distributions (one for each class) with two dimensional feature  
 415 vectors. The class-conditional densities  $p(x|y)$  have the same variance on the  
 416  $y$ -axis, but are horizontally elongated. Figure 2 illustrates an example of the  
 417 data distribution.

418 We investigate the abilities of approaches to learn one isotropic Gaussian  
 419 distribution per class. This model doesn't capture the horizontal elongation  
 420 of the true class distributions, so this forms a model mis-specification. Pa-  
 421 rameters of the model are the means and variances of Gaussian distributions.  
 422 To simplify the approach, we assume uniform prior probability for each class.

423 The training data set consists of 200 instances per class, where only few of  
 424 them are labeled and the testing data set consists of 200 instances per class.  
 425 Each experiment is run with different random initialization. The parameters  
 426 of the model are initialized by setting the means of the isotropic Gaussians  
 427 to the mean of the labeled instances, and setting the variances to one.

428 We run 150 experiments with 2, 4, and 6 labeled points, 50 runs for each,  
 429 where in each run we test different values of the hyper-parameters of each  
 430 method corresponding to the degree of importance of the unlabeled data (e.g.  
 431  $\gamma$  in IHA).

432 First, we investigate results of the Iterative Hybrid Algorithm. Figure 3  
 433 shows the performances achieved by the generative and discriminative mod-  
 434 els (we use the *MaxOne* variant of our approach) as a function of iteration  
 435 number in three particular runs, for different values of  $\gamma$ , where models are  
 436 trained on four labeled points during 50 iterations. They show typical behav-  
 437 iors of the method. Although supervised performances are often improved by

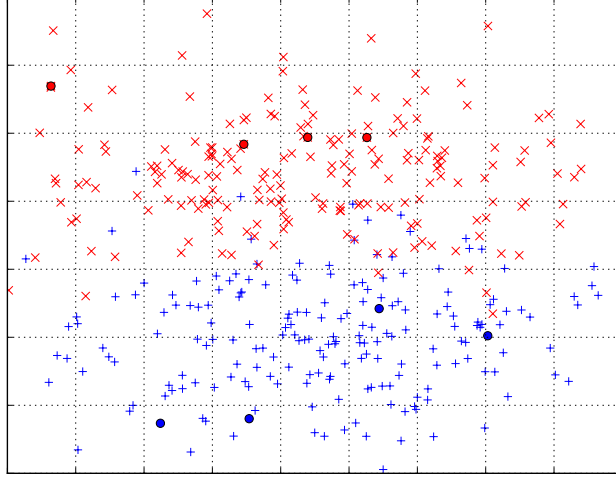


Figure 2: Example of the data distribution with four labeled points per class. The crosses correspond to unlabeled points and the circles correspond to the labeled points.

our iterative framework, performances might be unstable and not always improved from one iteration to another one. For instance, in figure *a* the method starts with a satisfying initial performance, but the performance drops with the number of iterations. In figure *b* the behavior is more chaotic and the accuracy decreases and increases twice, but the final performance achieved is better than the pure discriminative model. Less chaotic behavior is shown in figure *c*, where there are several small fluctuations of the performance, again resulting in better final performance than the pure discriminative model.

Then, we compare the performances of IHA, Hybrid Model and Entropy Minimization in supervised cases (setting hyperparameter values to particular values) and semi-supervised cases (looking for the best value of the hyperparameter). Figure 4 illustrates average performances where for each

method we provide on the left the performance of the supervised case and on the right the performance of the semi-supervised framework reached with the best hyperparameter value. In the Iterative Hybrid Algorithm,  $\lambda = 0$  corresponds to a discriminative model trained only on labeled data, but regularized with a generative model trained on both labeled and unlabeled data (see section 2.2). In the Entropy Minimization and the Hybrid Model  $\lambda = 0$  and  $\alpha = 1$ , respectively, correspond to pure discriminative model trained only on labeled data.

In all cases the usage of unlabeled data improves the performance. When there are only two labeled points the best performance is achieved by the Hybrid Model. On four and six labeled points, both the Iterative Hybrid Algorithm and the Hybrid model achieve high performance where the Hybrid Model slightly dominates on four labeled points and the Iterative Hybrid Algorithm dominates when six points are labeled. As it can be seen in the

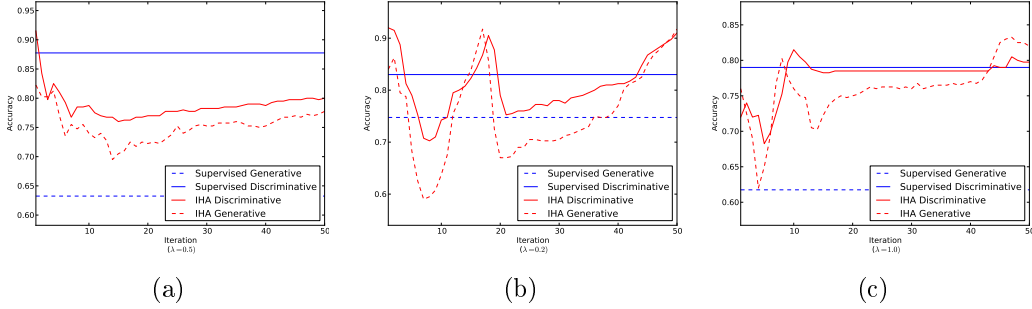


Figure 3: Performances of Iterative Hybrid Algorithm and of supervised training as a function of iteration number, as observed in three different runs for different values of  $\gamma$ . All experiments are performed on data with four labeled points.



figure, the difference in the performance for  $\lambda = 0$  and the best performance is not very large. This is due to the fact that although the discriminative model does not directly use the unlabeled data, its parameters are regularized with a generative model trained on both labeled and unlabeled data.

In table 1 we take a closer look at the best performance achieved by the methods in each run. We show the percentage of runs in which one method outperforms the other. Note that the numbers do not always sum up to 100% as in some cases the same performance is achieved by both methods. As in the previous figure, we may notice that on two labeled points the Hybrid Model is dominant. On four labeled points, however, the Iterative Hybrid Algorithm and the Hybrid Model achieve similar performance and

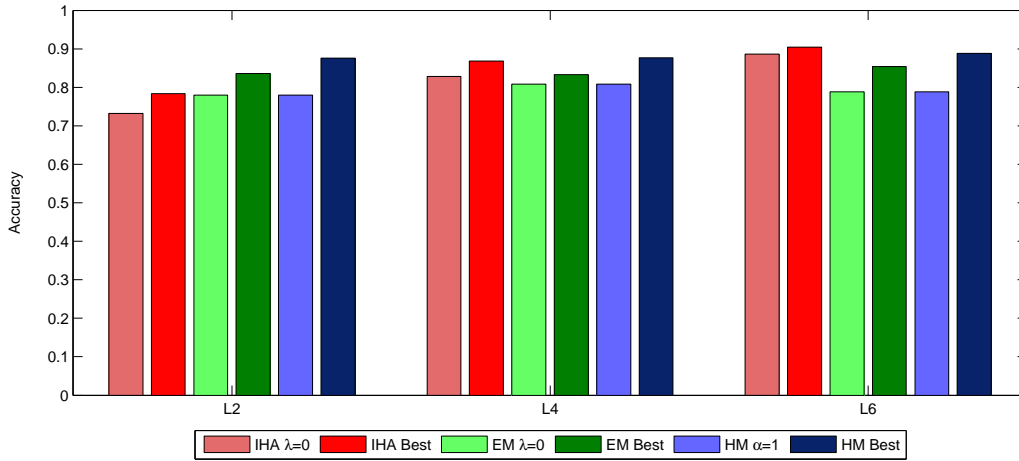


Figure 4: Average performances of the Iterative Hybrid Algorithm (in red), the Entropy Minimization method (in green) and the Hybrid Model (in blue) for supervised cases ( $\lambda = 0$  or  $\alpha = 1$  cases) and for semi-supervised cases (Best bar). Results are reported for 2 (left), 4 and 6 (right) labeled samples.

	L2				L4				L6		
	IHA	HM	EM		IHA	HM	EM		IHA	HM	EM
IHA	0%	16%	36%	IHA	0%	40%	72%	IHA	0%	72%	96%
HM	84%	0%	68%	HM	52%	0%	82%	HM	22%	0%	72%
EM	64%	24%	0%	EM	28%	14%	0%	EM	2%	16%	0%

Table 1: Percentage of runs in which the method on the left performs better then the method on the top, for 2, 4, and 6 labeled points.

share almost the same percentage of cases in which one outperforms the other. On the other hand, on six labeled points there is clear dominance of the Iterative Hybrid Algorithm, performing better than the Hybrid Model in 72% of the cases and better than the Entropy Minimization in 96% of the cases. In all cases the Entropy Minimization is outperformed by one of the other two methods.

## 6. Experiments on real datasets

### 6.1. Datasets and settings

*Datasets.* We experimented our SSL approach on financial time series and on handwriting data. We present these datasets now.

The financial time series dataset is composed of chart patterns which are particular shape of stock exchange series of interest for financial operators (see Figure 5). We used two databases of chart patterns, the first one (*CP4*) includes 448 series corresponding to the 4 most popular patterns *Head and Shoulders*, *Double Top*, *Reverse Head and Shoulders* and *Reverse Double Top*. The second dataset *CP8* includes 896 patterns from 8 classes, the four previous ones and four additional chart patterns : *Triple Top* (and the reverse

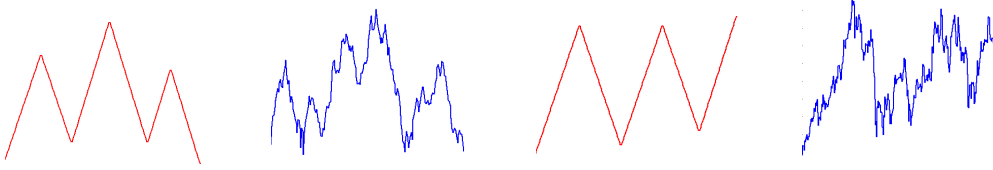


Figure 5: From left to right: ideal shape of a Head and Shoulder pattern (HS), observed HS, ideal shape of an Ascending Triangle pattern (AT), observed AT.

pattern), *Ascending Triangle* and *Descending Triangle*. Datasets are divided into 3 parts : a training dataset with 70 samples per class, a validation dataset and a test dataset with 20 samples per class each.

The handwriting database is a subset of the benchmark IAM database [49] which is made up of images of handwritten letters. The IAM database contains images of English handwriting words which are transformed into series of feature vectors by using a sliding window moving from the left to the right of the image and using preprocessing as in [49] to compute a feature vector from a position of the window. We used two versions of the dataset. A small dataset includes 23 classes and is divided into a training set with 200 samples per class, a validation set and a test set with 50 samples per class each. A bigger dataset includes 20 classes only (less represented classes have been removed) and consists of 2 600 samples per class in the training set and 600 samples per class in the validation set and in the test set.

*Benchmark methods.* We compared our approach with HMMs and HCRFs models trained in supervised setting and with HMMs and HCRFs trained in semi-supervised settings with state of the art approaches. We extend to

510 HCRFs two main semi-supervised approaches for CRFs which are the semi-  
511 supervised method based on entropy minimization [24], and the weighted  
512 method proposed in [25]. In the experiments, every HCRF training is per-  
513 formed through two steps: initialization by a HMM system and optimization  
514 using stochastic gradient descent on a regularized loss. Also we compared  
515 our approach to the general co-training algorithm where the two systems  
516 (HMMs and HCRFs) are trained on the same view.

517 In the following, we will name *HCRF init HMM* a HCRF system initial-  
518 ized by a HMM system, and respectively *weighted HCRF* a semi-supervised  
519 HCRF system based on the weighted approach in [25] and *entropy HCRF* a  
520 semi-supervised HCRF system based on the approach in [24].

521

522 *Experimental settings.* In all experiments on chart pattern classification the  
523 HMM and the HCRF models of every class are left-right models which have  
524 either 4 or 6 states depending on the shape of the figures (see Figure 5,  
525 e.g. the model of *Head and Shoulders* has six states since its composed of  
526 six segments). On IAM dataset all HMM and HCRF models are left-right  
527 models with 8 states.

528 One considered 50 samples of chart pattern training datasets as unlabeled  
529 data and for Small IAM and Big IAM datasets one uses 150 and 500 training  
530 samples as unlabeled data.

531 In all experiments, HMMs have a single Gaussian distribution with full  
532 covariance matrix as emission probability density.

533 For strengthening our results, we performed cross validation where folds  
534 are build on the training sets. In the following sections we provide first

preliminary results gained with 4 folds cross validation while in the final results (section 6.2.2), we performed 20 folds cross validation on IAM datasets and 60 folds cross validation on chart pattern datasets.

Note that it is not common to use a validation set in a semi-supervised learning because labeled samples are very few and more useful in the training set. Thus, in experiments below, the training of each model is performed through a fixed number of iterations, either 4 or 30, this is specified in the text.

## 6.2. Results

### 6.2.1. Preliminary results

*Iterative framework strategies.* We compare first the performances of the variants of our approach as detailed in section 4.3. Table 2 resumes the results.

database	iterations	Supervised		AllClasses		MaxProb		WeightOne		SelectProb	
		HMM	HCRF	HMM	HCRF	HMM	HCRF	HMM	HCRF	HMM	HCRF
CP4	4	77.2%	79.1%	78.4%	79.4%	<b>85.3%</b>	84.4%	85.0%	84.1%	78.8%	80.0%
	30	77.5%	78.8%	78.4%	79.1%	<b>85.0%</b>	84.7%	84.4%	84.1%	80.9%	80.6%
CP8	4	62.0%	61.1%	61.6%	61.6%	63.4%	<b>64.2%</b>	62.7%	64.1%	62.7%	64.1%
	30	62.5%	63.4%	62.7%	63.4%	<b>66.9%</b>	<b>66.9%</b>	66.1%	65.9%	64.2%	63.4%
small IAM	4	36.9%	38.9%	37.8%	39.0%	40.4%	41.5%	40.2%	<b>41.6%</b>	37.9%	39.3%
	30	37.2%	38.7%	37.1%	38.4%	39.4%	<b>40.1%</b>	39.1%	39.3%	38.2%	38.4%

Table 2: Performances on the test set of supervised HMM and supervised HCRF models compared to different variants of our approach.

The results are reported for 5 labeled data per class whatever the dataset, CP4, CP8 and Small IAM. Supervised models are trained either 4 or 30 iterations while every iteration of our iterative algorithm, models are retrained

551 for 4 iterations. This table shows that the *MaxProb* and the *WeightOne*  
 552 variants are often close and provide the best results while the *SelectProb* and  
 553 the *AllClasses* strategies are less efficient, especially the *AllClasses* strategy  
 554 which sometimes degrade the supervised case. Actually the *AllClasses* strat-  
 555 egy is very close to the *mixture* semi-supervised framework for HMMs, so  
 556 that this results confirm comments such as [19], [20] which conclude that  
 557 this method could sometimes decrease supervised performances. One note  
 558 also that in most cases running 30 training iterations degrades performances  
 559 : the labeled dataset size is too small so that models overfit. At the end,  
 560 one sees that the *MaxProb* strategy significantly outperform the purely su-  
 561 pervised training, for both HMM and HCRFs, and seems the best method  
 562 among all variants, when using only 4 training iterations. We will focus on  
 563 this variant only for the next experiments.

564

565 *Labeled datasets sizes.* Table 3 reports the performances of supervised HMM  
 566 and HCRF models compared to our iterative proposal for different number  
 567 of labeled data per class. We used from 1 to 10 labeled samples per class  
 568 and 50 or 150 unlabeled samples per class according to the database. One  
 569 can see that whatever the dataset, CP8 and Small IAM, HCRFs outperform  
 570 corresponding HMMs and that our approach systematically and significantly  
 571 outperform the supervised setting.

572 *Evolution of the performances during training.* It is interesting to look at  
 573 the evolution of the performance as a function of the iteration number in  
 574 our iterative algorithm. Figure 6 plots the performances of HCRF systems  
 575 and of HMM systems as a function of the iteration number for two iterative

labeled data	CP8				Small IAM			
	Supervised		IHA		Supervised		IHA	
	HMM	HCRF	HMM	HCRF	HMM	HCRF	HMM	HCRF
1	32.5%	38%	48.9%	49.4%	14.7%	19.1%	23.7%	23.9%
2	51.4%	51.4%	55.9%	56.7%	24.6%	28.5%	30.0%	30.7%
5	62.0%	61.1%	63.4%	64.2%	36.9%	38.9%	40.4%	41.5%
10	62.7%	63.9%	66.3%	66.6%	46.3%	47.1%	47.1%	48.1%

Table 3: Performances on test set of supervised training for HMMs and HCRFs, compared to semi-supervised training with our approach (*MaxProb* variant) as a function of the number of labeled samples per class, while the number of unlabeled samples remain fixed to 50 samples per class for the CP8 database and 150 samples per class for the Small IAM database.

576 algorithms, the co-training algorithm and our IHA approach. Note that  
577 here one iteration stands for a retraining of both the HMM system and the  
578 HCRF system. In our approach this corresponds to a retraining of the HMM  
579 systems based on HCRFs classification of unlabeled data and a retraining of  
580 the HCRF from the HMM solution. We used here 10 labeled samples and  
581 still 50 or 150 unlabeled samples per class for CP8 and Small IAM databases.

582 We plot as a reference the performance of supervised learning, both for  
583 HMMs and for HCRFs. One sees that both iterative algorithms allow improv-  
584 ing over supervised training with our proposal being slightly more efficient  
585 than standard co-training. The performance of both HMMs and HCRFs in-  
586 crease almost monotonously until it converges. Note also that our approach  
587 may reach its best results after few iterations (Small IAM dataset) or may  
588 require more iteration (CP8 dataset) to converge to an accurate solution, it

589 depends on the datasets.

590 *Influence of unlabeled data.* At last we investigated the influence of the num-  
591 ber of unlabeled data on the performance of our approach. Figure 7 shows  
592 the evolution of the accuracy of few systems as the number of unlabeled sam-  
593 ples increases while the number of labeled samples remains fixed (5 samples  
594 per class). We compare systems learned in a purely supervised settings, semi-  
595 supervised systems, i.e. a HMM system learned with a mixture strategy and  
596 a HCRF system learned in a supervised setting from this HMM solution (as  
597 discussed in section 2.2), and systems learned using our iterative approach  
598 (note that the semi-supervised systems correspond to the systems gained at  
599 the end of the first iteration of our hybrid algorithm). We used 5 labeled  
600 samples per class and from 25 to 500 unlabeled samples per class. The first  
601 point to note is that SSL systems (both HMMs and HCRFs) outperform  
602 the corresponding supervised systems and that the accuracy increases up to  
603 a plateau. The second point is that the iterative algorithm allows learn-  
604 ing even more accurate classifiers, both HMMs and HCRFs, and that their  
605 performance increases steadily with the number of unlabeled data.

#### 606 6.2.2. Comparative results with state of the art methods

607 Finally, we compared more extensively our methods with more state of  
608 the art semi-supervised methods on our four datasets (CP4, CP8, small IAM  
609 and Big IAM). In these experiments to provide more significant results we  
610 report averaged results gained with 20 folds cross validation on the IAM  
611 corpus and with 60 folds cross validation on Chart Pattern databases. This  
612 allows providing 95% confidence interval. Note that since standard semi-



supervised training of HMM do not always improve over the supervised case,  
 we investigated different variants of the standard algorithm and obtained that  
 semi-supervised HMMs offers best performances when using the *SelectProb*  
 and the *MaxProb* strategies as in equations 20 and 18, we report here best  
 results following datasets that is *MaxProb* strategy for Chart Patterns and  
*SelectProb* for IAM datasets. Also, following preliminary results, we chose to  
 use the *MaxProb* strategy for our iterative framework. All models are trained  
 on 5 labeled samples per class and we use 50 unlabeled samples per class on  
 the Chart Pattern datasets, 150 and 500 unlabeled samples per class on the  
 Small IAM and Big IAM datasets.

method	CP4	CP8	Small IAM	Big IAM
Supervised HMM	78.5% $\pm$ 1.1	59.3% $\pm$ 0.9	35.8% $\pm$ 1.0	40.9% $\pm$ 0.9
Supervised HCRF	78.7% $\pm$ 1.1	59.7% $\pm$ 0.9	37.6% $\pm$ 1.0	42.0% $\pm$ 0.9
SSL HMM	83.8% $\pm$ 0.6	61.8% $\pm$ 0.9	36.6% $\pm$ 1.2	42.7% $\pm$ 1.0
SSL HCRF init HMM	83.9% $\pm$ 0.6	62.0% $\pm$ 1.00	37.6% $\pm$ 1.2	43.2% $\pm$ 1.0
SSL HCRF entropy	84.0% $\pm$ 0.6	62.0% $\pm$ 0.9	37.6% $\pm$ 0.9	43.2% $\pm$ 1.0
SSL HCRF weighted	83.9% $\pm$ 0.5	62.0% $\pm$ 0.9	37.7% $\pm$ 0.9	43.2% $\pm$ 1.0
Co-training HMM	83.5% $\pm$ 0.6	61.5% $\pm$ 0.9	35.7% $\pm$ 0.9	40.9% $\pm$ 0.9
Co-training HCRF	83.5% $\pm$ 0.7	61.9% $\pm$ 0.9	<b>39.5%</b> $\pm$ 0.9	43.6% $\pm$ 0.9
IHA HMM	84.0% $\pm$ 0.5	62.1% $\pm$ 0.9	38.8% $\pm$ 1.0	44.1% $\pm$ 0.9
IHA HCRF	<b>84.2%</b> $\pm$ 0.5	<b>62.4%</b> $\pm$ 0.9	38.9% $\pm$ 1.0	<b>44.5%</b> $\pm$ 0.9

Table 4: Comparison of our proposed semi-supervised HCRF and iterative  
 framework with state of the art methods : Semi-supervised learning of HCRFs  
 using entropy or weighted approaches and the general co-training algorithm.

This table calls from a few comments. First SSL learning systematically

624 outperform supervise learning, for both HMM and HCRF systems. Second,  
625 entropy SSL entropy and weighted SSL are very close whatever the dataset,  
626 and yield small improvement over simple SSL training for HMMs and for  
627 HCRFs, if any. Co-training performs sometimes better, and sometimes worse  
628 than simple SSL methods, and appears to be less robust and maybe more  
629 difficult to tune. Finally, our Iterative Hybrid Algorithm most often outper-  
630 forms all other methods, both for HMM and for HCRF systems. Note that  
631 although improvements are often small, they are bigger on the IAM data set  
632 than on the CPx datasets and they are systematic.

633 As a conclusion our approach allows exploiting unlabeled data to improve  
634 the behavior of both generative systems and of discriminative ones. It com-  
635 pares well to state of the art methods for SSL learning and most often leads  
636 to best results for the discriminative system. Importantly a by-product of  
637 the algorithm is an efficient SSL trained generative system which significantly  
638 outperform other SSL learning for these models.

## 639 7. Conclusion

640 We presented a joint HMM-HCRF framework for semi-supervised learn-  
641 ing of graphical models for sequences. Our approach combines on the one  
642 hand the initialization scheme of HCRFs system by HMMs for learning a  
643 HCRF model using unlabeled data and on the other hand a co-training pro-  
644 cedure for improving a HMM system based on a HCRF model. Our ex-  
645 perimental results on two datasets show that our strategy efficiently allows  
646 taking into account unlabeled data both for learning the discriminative mod-  
647 els (HCRF) and the generative models (HMMs). It compares well to state

648 of the art semi-supervised approaches applied to HCRF learning and to the  
649 well known co-training algorithm.

## 650 References

- 651 [1] L. R. Rabiner, A tutorial on hidden markov models and selected appli-  
652 cations in speech recognition, in: Proceedings of the IEEE, 1989, pp.  
653 257–286.
- 654 [2] B. Juang, S. Katagiri, Discriminative learning for minimum error clas-  
655 sification, in: IEEE Trans. Acoustics, Speech, and Signal Processing,  
656 Vol.40, No.12, 1992.
- 657 [3] M. Collins, Discriminative training methods for hidden markov models:  
658 theory and experiments with perceptron algorithms, in: EMNLP, 2002.
- 659 [4] P. Woodland, D. Povey, Large scale discriminative training of hidden  
660 markov models for speech recognition, Computer Speech and Language.
- 661 [5] F. Sha, L. K. Saul, Large margin hidden markov models for automatic  
662 speech recognition, in: B. Scholkopf, J. Platt, T. Hoffman (Eds.), in Ad-  
663 vances in Neural Information Processing Systems 19, MIT Press, 2007,  
664 pp. 1249–1256.
- 665 [6] T.-M.-T. Do, T. Artières, Large margin training for hidden Markov  
666 models with partially observed states, in: L. Bottou, M. Littman (Eds.),  
667 Proceedings of the 26th International Conference on Machine Learning,  
668 Omnipress, Montreal, 2009, pp. 265–272.

- 669 [7] J. D. Lafferty, A. McCallum, F. C. N. Pereira, Conditional random  
670 fields: Probabilistic models for segmenting and labeling sequence data,  
671 in: Proceedings of the Eighteenth International Conference on Machine  
672 Learning, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA,  
673 2001, pp. 282–289.
- 674 [8] A. Gunawardana, M. Mahajan, A. Acero, J. C. Platt, Hidden conditional  
675 random fields for phone classification, in: Interspeech, 2005, pp. 1117–  
676 1120.
- 677 [9] S. Reiter, B. Schuller, G. Rigoll, Hidden conditional random fields for  
678 meeting segmentation, in: ICME, 2007, pp. 639–642.
- 679 [10] T.-M.-T. Do, T. Artières, Conditional random field for tracking user  
680 behavior based on his eye’s movements, in: NIPS’05 Workshop on Ma-  
681 chine Learning for Implicit Feedback and User Modeling, Whistler, BC,  
682 Canada, 2005.
- 683 [11] T.-M.-T. Do, T. Artières, Conditional Random Fields for Online Hand-  
684 writing Recognition, in: Guy Lorette (Ed.), Tenth International Work-  
685 shop on Frontiers in Handwriting Recognition, Université de Rennes 1,  
686 Suvisoft, La Baule (France), 2006.
- 687 [12] A. Vinel, T. M. T. Do, T. Artières, Joint optimization of hidden con-  
688 ditional random fields and non linear feature extraction, in: ICDAR,  
689 2011, pp. 513–517.
- 690 [13] L.-P. Morency, A. Quattoni, T. Darrell, Latent-dynamic discriminative  
691 models for continuous gesture recognition, in: CVPR, 2007.

- [14] Y. Soullard, T. Artieres, Hybrid hmm and hcrf model for sequence classification, in: Proceedings of the European Symposium on Artificial Neural Networks (ESANN 2011), Computational Intelligence and Machine Learning. Bruges, Belgium, 2011, pp. 453–458.
- [15] Y. Grandvalet, Y. Bengio, Semi-supervised learning by entropy minimization, *Network* 17 (5) (2005) 529–536.
- [16] J. Wang, X. Shen, W. Pan, On efficient large margin semisupervised learning: Method and theory, *Journal of Machine Learning Research* 10 (2009) 719–742.
- [17] A. Blum, T. Mitchell, Combining labeled and unlabeled data with co-training, Morgan Kaufmann Publishers, 1998, pp. 92–100.
- [18] G. S. Mann, A. McCallum, Generalized expectation criteria for semi-supervised learning with weakly labeled data, *Journal of Machine Learning Research* 11 (2010) 955–984.
- [19] K. Nigam, A. McCallum, S. Thrun, T. M. Mitchell, Text classification from labeled and unlabeled documents using em, *Machine Learning* 39 (2/3) (2000) 103–134.
- [20] M. Inoue, N. Ueda, Exploitation of unlabeled sequences in hidden markov models, *IEEE Trans. On Pattern Analysis and Machine Intelligence* 25 (2003) 1570–1581.
- [21] G. Haffari, A. Sarkar, Homotopy-based semi-supervised hidden markov models for sequence labeling, in: COLING, 2008, pp. 305–312.

- 714 [22] F. G. Cozman, I. Cohen, Unlabeled data can degrade classification per-  
715 formance of generative classifiers, in: Fifteenth International Florida  
716 Artificial Intelligence Society Conference, 2002, pp. 327–331.
- 717 [23] B. Merialdo, Tagging english text with a probabilistic model, *Compu-*  
718 *tational Linguistics* 20 (2) (1994) 155–171.
- 719 [24] F. Jiao, Semi-supervised conditional random fields for improved se-  
720 quence segmentation and labeling, in: In COLING/ACL, COL-  
721 ING/ACL, 2006, pp. 209–216.
- 722 [25] N. Sokolovska, Aspects of semi-supervised and active learning in condi-  
723 tional random fields, in: ECML/PKDD (3), 2011, pp. 273–288.
- 724 [26] Y. Soullard, T. Artieres, Iterative refinement of hmm and hcrf for  
725 sequence classification, in: IAPR Workshop on Partially Supervised  
726 Learning (PSL), 2011.
- 727 [27] A. Quattoni, S. B. Wang, L.-P. Morency, M. Collins, T. Darrell, Hidden-  
728 state conditional random fields, *IEEE Transactions on Pattern Analysis*  
729 *and Machine Intelligence* 29 (10) (2007) 1848–1852.
- 730 [28] G. S. N. W. L. E. Baum, T. Petrie, A maximization technique occurring  
731 in the statistical analysis of probabilistic functions of markov chains  
732 (1970) 164–171, vol 41, No 1.
- 733 [29] L. Bottou, On-line learning in neural networks, Cambridge University  
734 Press, New York, NY, USA, 1998, Ch. On-line learning and stochastic  
735 approximations, pp. 9–42.

- 736 [30] D. C. Liu, J. Nocedal, D. C. Liu, J. Nocedal, On the limited memory  
737 bfgs method for large scale optimization, *Mathematical Programming*  
738 45 (1989) 503–528.
- 739 [31] F. Sha, Large margin training of acoustic models for speech recognition,  
740 Doctoral dissertation (2006).
- 741 [32] U. Brefeld, T. Scheffer, Semi-supervised learning for structured output  
742 variables, in: *ICML*, 2006, pp. 145–152.
- 743 [33] X. Zhu, Semi-Supervised Learning Literature Survey Contents, Sciences-  
744 New York.
- 745 [34] S. Baluja, Probabilistic modeling for face orientation discrimination:  
746 Learning from labeled and unlabeled data, *Advances in Neural Infor-*  
747 *mation Processing Systems* 11 (1998) 854–860.
- 748 [35] W. Wang, Z. hua Zhou, Analyzing co-training style algorithms, in: *Pro-*  
749 *ceedings of the 18th European Conference on Machine Learning*, 2007.
- 750 [36] D. Zhou, J. Huang, B. Scholkopf, Learning from Labeled and Unlabeled  
751 Data on a Directed Graph, *Proceedings of the 22nd International Con-*  
752 *ference on Machine Learning* (2005) 1041–1048.
- 753 [37] A. Blum, S. Chawla, Learning from Labeled and Unlabeled Data using  
754 Graph Mincuts, in: *Science*, Morgan Kaufmann Publishers Inc., 2001,  
755 pp. 19–26.
- 756 [38] E. Riloff, J. Wiebe, Learning Extraction Patterns for Subjective Expres-  
757 sions, in: E. Riloff, J. Wiebe (Eds.), *Proceedings of the Conference on*

- 758 Empirical Methods in Natural Language Processing EMNLP, Confer-  
759 ence on Empirical Methods in Natural Language Processing, 2003, pp.  
760 105–112.
- 761 [39] D. Yarowsky, Unsupervised word-sense disambiguation rivalling super-  
762 vised methods, in: 33rd Annual Meeting of the Association for Compu-  
763 tational Linguistics, 1995, pp. 189–196.
- 764 [40] J. L. Christopher M. Bishop, Generative or discriminative? getting the  
765 best of both worlds (2006) 3–24.
- 766 [41] G. Bouchard, Bias-variance tradeoff in hybrid generative-discriminative  
767 models, in: ICMLA '07: Proceedings of the Sixth International Confer-  
768 ence on Machine Learning and Applications, IEEE Computer Society,  
769 Washington, DC, USA, 2007, pp. 124–129.
- 770 [42] S. Ji, L. T. Watson, L. Carin, Semisupervised learning of hidden markov  
771 models via a homotopy method., IEEE Trans. Pattern Anal. Mach. In-  
772 tell. 31 (2) (2009) 275–287.
- 773 [43] Y. Wang, G. Haffari, S. Wang, G. Mori, A rate distortion approach for  
774 semi-supervised conditional random fields, in: NIPS, 2009, pp. 2008–  
775 2016.
- 776 [44] J. Suzuki, A. Fujino, H. Isozaki, Semi-supervised structured output  
777 learning based on a hybrid generative and discriminative approach, in:  
778 EMNLP-CoNLL, 2007, pp. 791–800.
- 779 [45] S. Z. K. Khine, T. L. Nwe, H. Li, Singing voice detection in pop songs  
780 using co-training algorithm., in: ICASSP, IEEE, 2008, pp. 1629–1632.



- 781 [46] V. Frinken, T. Peter, A. Fischer, H. Bunke, T.-M.-T. Do, T. Artieres,  
782 Improved handwriting recognition by combining two forms of hidden  
783 markov models and a recurrent neural network, in: CAIP '09: Pro-  
784 ceedings of the 13th International Conference on Computer Analysis  
785 of Images and Patterns, Springer-Verlag, Berlin, Heidelberg, 2009, pp.  
786 189–196.
- 787 [47] A. Y. Ng, M. I. Jordan, On discriminative vs. generative classifiers: A  
788 comparison of logistic regression and naive bayes, in: NIPS, 2001, pp.  
789 841–848.
- 790 [48] J. Lasserre, Hybrid of generative and discriminative methods for ma-  
791 chine learning (thesis).
- 792 [49] U. Marti, H. Bunke, A full english sentence database for off-line hand-  
793 writing recognition, in: ICDAR, 2002.

## 794 **Appendix A. HCRF initialization from HMMs**

795 We consider here HMMs with one Gaussian distribution with full covari-  
796 ance matrix as emission probability density. We show how a HMM of  $Q$   
797 states may be used to initialize a HCRF with the same topology.

We note for any state  $i \in [1, Q]$ ,  $\boldsymbol{\mu}^i$  and  $\Sigma^i$  the mean and covariance matrix of the gaussian distribution in state  $i$ . Let  $d$  the dimension of feature vectors, then  $\boldsymbol{\mu}^i$  is a vector of dimension  $d$  and  $\Sigma^i$  is a matrix of dimension

$d \times d$ . Then, we define the following feature maps and parameter vectors:

$$\begin{aligned}\phi^{trans}(\mathbf{x}, y, h_t, h_{t-1}) &= (\delta_{h_t=1 \wedge h_{t-1}=1}, \dots, \delta_{h_t=Q \wedge h_{t-1}=Q})^T \\ \lambda^{trans} &= (\log a_{1,1}, \dots, \log a_{Q,Q})^T \\ \phi^{loc}(\mathbf{x}, y, h_t) &= (\phi_1^{loc}(\mathbf{x}, y, h_t), \phi_2^{loc}(\mathbf{x}, y, h_t), \dots, \phi_Q^{loc}(\mathbf{x}, y, h_t))^T \\ \lambda^{loc} &= (\lambda_1^{loc}, \lambda_2^{loc}, \dots, \lambda_Q^{loc})^T\end{aligned}$$

798 where  $a_{i,j}$  stands for the usual HMM transition probability from state  $i$  to  $j$ .

799

Using these definition, we get easily that if  $\phi^{loc}$  and  $\lambda^{loc}$  satisfy:

$$\langle \lambda_i^{loc}, \phi_i^{loc}(\mathbf{x}, y, s_i) \rangle = (\mathbf{x} - \boldsymbol{\mu})^T (\Sigma^i)^{-1} (\mathbf{x} - \boldsymbol{\mu}) - \frac{1}{2} \log((2\pi)^d |\Sigma^i|) \quad (\text{A.1})$$

then:

$$e^{\lambda_i^{loc} \cdot \phi_i^{loc}(\mathbf{x}, y, h_t)} = \frac{1}{\sqrt{(2\pi)^d |\Sigma^i|}} e^{-\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^T (\Sigma^i)^{-1} (\mathbf{x} - \boldsymbol{\mu})} \quad (\text{A.2})$$

800

801

It is easy to check that the definitions below allow satisfying the equality above:

$$\begin{aligned}\phi_i^{loc}(\mathbf{x}, y, h_t) &= \left(1, (x_u)_u, (x_u x_v)_{u,v}\right)^T \times \delta_{h_t=i} \quad \forall i \in [1, Q] \\ \lambda_i^{loc} &= \begin{pmatrix} -\frac{1}{2} \left[ \log((2\pi)^d |\Sigma^i|) + (\boldsymbol{\mu}^i)^T (\Sigma^i)^{-1} \boldsymbol{\mu}^i \right] \\ (\Sigma^i)^{-1} \boldsymbol{\mu}^i \\ \left(-\frac{1}{2} (\Sigma^i)^{-1}\right)_{u,v} \end{pmatrix} \quad \forall i \in [1, Q]\end{aligned}$$

802

803

where we use the notation:

$$(x_u)_u = \mathbf{x} = (x_1, \dots, x_d)^T$$

$$(x_u x_v)_{u,v} = \mathbf{x} \otimes \mathbf{x} = (x_1^2, x_1 x_2, \dots, x_1 x_d, x_2 x_1, \dots, x_d x_1, \dots, x_d^2)^T$$

804 In previous derivation  $A_{u,v}$  stands for the element line  $u$  and column  $v$  of  
 805 the matrix  $A$ . Also,  $(A_{u,v})_{u,v} = (A_{1,1}, A_{1,2}, \dots, A_{2,1}, A_{2,2}, \dots, A_{3,1}, \dots)^T$  is a  
 806 vector of the elements of  $A$  unfolded in column first order.

807

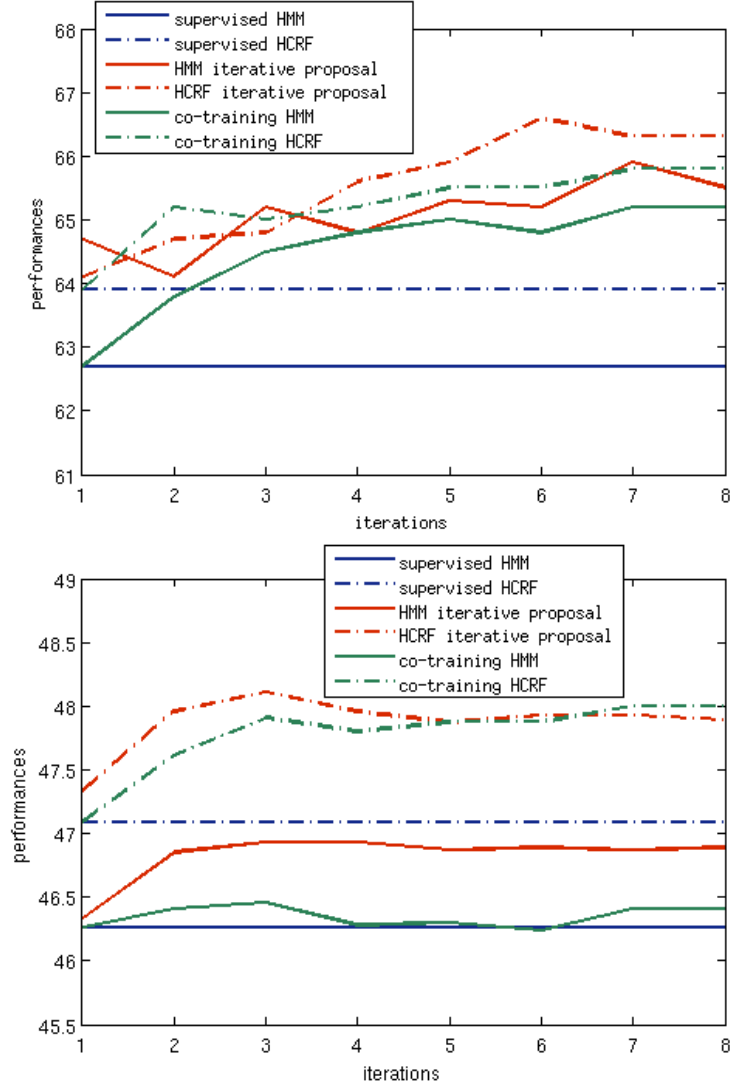


Figure 6: Performance on the test set of HMMs and HCRFs models following iteration number in our iterative hybrid algorithm for CP8 (top) and small IAM (bottom) datasets.

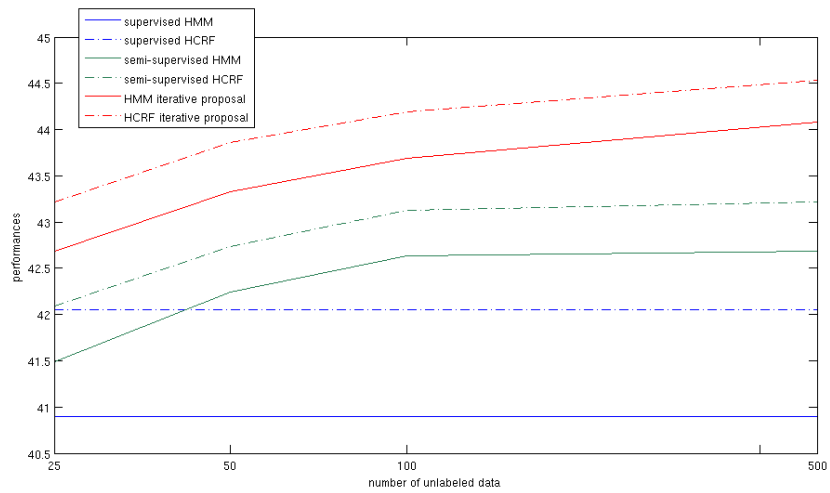


Figure 7: Comparison of the accuracy of HMM and HCRF systems trained in a supervised and in a semi-supervised setting with HMM and HCRF systems learned with our iterative approach on the Big IAM dataset. Performance is plotted as a function of the number of unlabeled samples used.