

International Journal of Pattern Recognition and Artificial Intelligence
© World Scientific Publishing Company

Semi-supervised Sequence Classification with HMMs

Shi Zhong

*Department of Computer Science and Engineering
Florida Atlantic University
777 Glades Rd, Boca Raton, Florida 33431, USA
zhong@cse.fau.edu
<http://www.cse.fau.edu/~zhong>*

Using unlabeled data to help supervised learning has become an increasingly attractive methodology and proven to be effective in many applications. This paper applies semi-supervised classification algorithms, based on hidden Markov models, to classify sequences. For model-based classification, semi-supervised learning amounts to using both labeled and unlabeled data to train model parameters. We examine three different strategies of using labeled and unlabeled data in the model training process. These strategies differ in how and when labeled and unlabeled data contribute to the model training process. We also compare regular semi-supervised learning, where there are separate unlabeled training data and unlabeled test data, with transductive learning, where we do not differentiate between unlabeled training data and unlabeled test data. Our experimental results on synthetic and real EEG time-series show that substantially improved classification accuracy can be achieved by these semi-supervised learning strategies. The effect of model complexity on semi-supervised learning is also studied in our experiments.

Keywords: Semi-supervised Learning; Sequence Classification; Hidden Markov Models.

1. Introduction

Learning with both labeled and unlabeled data has been studied with great interest recently. Though theoretical justification on the value of unlabeled data has not been promising [5, 27], novel semi-supervised algorithms and successful applications are abundant [4, 12, 19, 26, 3, 1]. For example, it has been shown that unlabeled data can significantly improve the classification accuracy or information retrieval performance in applications such as text classification [12, 19], terrain classification [9], gesture recognition [26], and content-based image retrieval [6]. Semi-supervised learning can also be viewed as using labeled data as feedback to help cluster unlabeled data. This leads to semi-supervised clustering; e.g., Basu et al. [1] studied the effectiveness of seeded k-means and constrained k-means as semi-supervised techniques for clustering text documents.

Semi-supervised learning was motivated by many real-world problems. For example, text categorization can be tedious for a human—one has to read through a document and put it in an appropriate predefined category. Many recent works [19, 17] aim to reduce the number of text documents to be labeled by a human

while achieving the same level of classification accuracy by exploiting information in unlabeled documents. In gene expression analysis, profound expert knowledge and costly biological experiments are often required to manually label the functional category of each gene. Semi-supervised learning methods can help reduce such efforts by automatically grouping unlabeled/unknown genes into meaningful categories based on a limited number of manually-classified genes.

With the same motivation, the intent of this paper is to classify sequence data with minimal human labeling efforts. Sequence classification problems have been encountered in real applications such as robot motion control, biosignal analysis [22, 29], and speech recognition [11]. The semi-supervised learning paradigm studied in this paper is similar to those in [19], [11], and [1].

The main contribution of this paper is an empirical study of semi-supervised learning of hidden Markov models (HMMs) for sequence classification. Through experiments on both synthetic and real EEG time-series data, we study three different strategies of combining labeled and unlabeled sequences for learning hidden Markov models. This paper extends our previous work [28] in several aspects: (a) we include for comparison a soft version of one of the three combining strategies and the results confirm that there is little difference between soft and hard version for learning complex models, which has theoretically been explained in [30]; (b) we differentiate two different semi-supervised learning paradigms (regular semi-supervised learning and *transductive learning* [12]) and compare them in our HMM-based classification experiments; (c) we conduct more experiments on larger datasets and study the effect of model complexity (i.e., different number of hidden states for hidden Markov models) on the performance of semi-supervised learning.

Although exploitation of unlabeled sequences for learning HMMs has been studied before [11], this paper has significant difference. Existing work [11] focused on empirically investigating the value of labeled and unlabeled sequences for training hidden Markov models and only one EM-based combining strategy was studied. On the other hand, two strategies of combining labeled and unlabeled data were empirically studied in [1], but for a document clustering problem. We not only appear to be the first to study these strategies for HMM-based sequence classification, but also have not seen any empirical work on comparing different semi-supervised learning paradigms. In this paper, by “sequence”, we refer to a sequence of numbers (discrete or continuous) at discrete time points. The time-series data type used in our experiments are a subtype of sequence data. Our methodology, however, applies to any sequence data that can be modeled by HMMs.

One benefit of using HMMs to model sequences is that there is no need to extract feature vectors and stationary temporal characteristics can be captured. Hidden Markov models have proven to be very effective in characterizing a wide variety of sequential data [20, 22]. For example, the real EEG time-series (used in our experiments) have been effectively classified using HMMs [29].

The organization of this paper is as follows. The semi-supervised sequence learning problem is formalized in the the next section, followed by an introduction to

HMMs. We then present three semi-supervised classification strategies and show improved classification results on two time-series datasets. Finally, we conclude this paper with remarks on related work and future work.

2. Semi-supervised Learning Problem

Here we make a distinction between two slightly different semi-supervised learning/classification paradigms—regular semi-supervised learning and transductive learning. The former uses a set of training data that contain both labeled and unlabeled data, and the trained classifier will then be used to classify future unlabeled data, thus an independent unlabeled test dataset is used to evaluate the performance of the trained classifier. The latter, in contrast, does not differentiate unlabeled training data from unlabeled test data. Any unlabeled data we want to classify can be added to help train the classifier. Both learning types seem to be useful, possibly for different practical situations though. The former is desirable if one does not want to keep labeled training data forever or to retrain the classification model whenever a new set of unlabeled data become available. On the other hand, if the training data are not representative enough and unlabeled test data have (slightly) different distributions, transductive learning may be able to help correct the classifier so that it captures better the characteristics of unlabeled test data and thus classifies more accurately.

The difference between these two learning paradigms, however, is not significant and is more conceptual than mathematical since the training process of the former can be viewed exactly as a transductive learning process. Let us formally describe the transductive learning as follows:

There are two types of data: labeled data $\mathcal{L} = \{o_i^{(l)}, y_i^{(l)}\}_{i=1}^{N_l}$ and unlabeled data $\mathcal{U} = \{o_i^{(u)}\}_{i=1}^{N_u}$, where $\{o\}$ represents a set of data instances, $\{y\}$ a set of class labels, N_l the number of labeled data instances, and N_u the number of unlabeled instances. The goal is to label the unlabeled instances based on all data instances.

This problem is general in that we do not know whether or not the class labels contained in labeled data are complete for unlabeled data. It is totally legal to discover new labels from unlabeled data, but can be very difficult. An extreme situation of the problem is the $N_l = 0$ case when one has to assign labels to all data. One then usually resorts to unsupervised clustering algorithms, by which “similar” data instances are grouped together and assigned a group label. This brings up a connection between semi-supervised classification and unsupervised clustering and thus one will not be surprised to see “soft” EM and “hard” k-means clustering coming into semi-supervised learning algorithms in Section 4. In this paper, we restrict ourselves to the scenario that labels contained in the labeled data are complete and we aim to assign these labels to each and every unlabeled data instance.

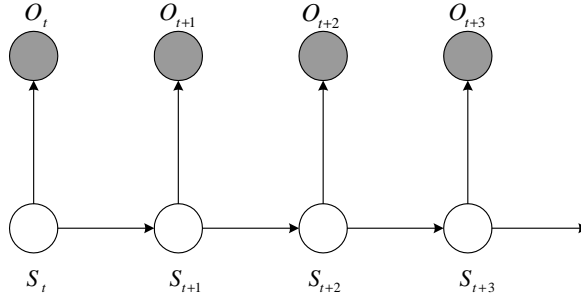


Fig. 1. A first order HMM model. The empty circles are hidden states and the shaded ones observations.

Our important assumption in this paper is that both labeled and unlabeled sequences are generated from the same set of models. That is, each class of sequences is modeled by an HMM λ and the likelihood $P(o|\lambda)$ used as a measure of how likely a sequence o is generated from the model λ . Our goal here is to use unlabeled sequences to aid the generation of better HMM models which in turn better classify the unlabeled sequences. Next we briefly describe hidden Markov models.

3. Hidden Markov Models

HMMs have been heavily researched and used for the past several decades, especially in the speech recognition area [20]. A standard HMM model uses a discrete hidden state at time t to summarize all the information before t and thus the observation at any time only depends on the current hidden state. The hidden state sequence is a Markov chain. In this paper we use the simplest HMM, a univariate first order HMM, in which the observation is a scalar at any time and the state sequence is a first order Markov chain. Such an HMM unrolled over several time slices is shown in Figure 1.

A standard HMM is usually denoted as a triplet $\lambda = (\pi, A, B)$. $\pi = \{\pi_i\}$ (where $\sum_i \pi_i = 1$) is the prior probability distribution of hidden states. $A = \{a_{ij}\}$ (where $\sum_j a_{ij} = 1$) is the transition probability distribution between hidden states. For discrete observation case, the observation distribution is $B = \{b_j(k)\}$ (where $\sum_k b_j(k) = 1$). For continuous observation case, the observation distribution is usually modeled by a mixture of Gaussians

$$b_j(o) = \sum_l c_{jl} \mathcal{N}[o, \mu_{jl}, U_{jl}] , \quad (1)$$

where $\sum_l c_{jl} = 1$, o is the observation vector being modeled, c_{jl} the mixture weight, μ_{jl} the mean vector of the m -th mixture, U_{jl} the covariance matrix of the l -th mixture for state j and \mathcal{N} is the Gaussian density function.

Three basic problems of interest for HMMs are: evaluating the likelihood $P(o|\lambda)$ of an observation sequence o given an HMM λ ; finding the most likely hidden state sequence S corresponding to an observation o given λ , and learning the parameters of a model λ given a set of observations O . The evaluation and learning of an HMM both exploit an efficient forward-backward inference procedure [2]. In semi-supervised HMM-based classification, we will encounter the first (for classifying a sequence) and the last (for estimating HMMs' parameters) problem. For simplicity, the mathematical formulae for parameter estimation of an HMM given a set of sequences and the likelihood calculation are omitted here and readers are referred to [20, 7] for more details.

4. Semi-supervised HMM-based Classification (SSHC)

As mentioned in Section 2, semi-supervised model-based classification problem is closely related to clustering, specifically, model-based partitional clustering [30], which involves two basic steps: (a) in a data assignment step each data instance is assigned to one or more model(s); and (b) in a model estimation step, one estimates the parameters of each model using the data instances assigned to it. In the first step, hard k-means assignment and soft EM assignment are two most popular choices, even though others exist (e.g., *stochastic* assignment, see [14, 30]).

In this paper, we mainly consider hard assignment, where each data instance is assigned to only one model/cluster, which results in a k-means type algorithm that has been used by many researchers [24, 15]. Compared to soft EM assignment [11], hard k-means seems to be computationally more efficient, yet produces similar results for complex models (such as HMMs) [30]. To demonstrate the similarity, we implement an EM version of the SSHC-KM2 algorithm presented next and show its performance in Section 5.3.

We construct semi-supervised classification algorithms by modifying the HMM-based k-means clustering algorithm to accommodate labeled sequences. The modification basically amounts to using labeled data to initialize HMMs and/or to constrain the parameter estimation of HMMs. The next section presents three different versions of semi-supervised HMM-based sequence classification algorithms that are based on hard k-means assignment. The first two versions can be seen as HMM-based extensions of the constrained k-means and seeded k-means algorithms proposed in [1]. The first one can also be regarded as a hard version of the algorithm studied in [11].

4.1. Semi-supervised Algorithms

The first version, we call *SSHC-KM1*, is shown in Figure 2. It is also the most straightforward combination of HMM-based k-means with supervised training—first initializing HMMs using labeled sequences and then iterating between two steps: labeling unlabeled sequences and updating models using both the original

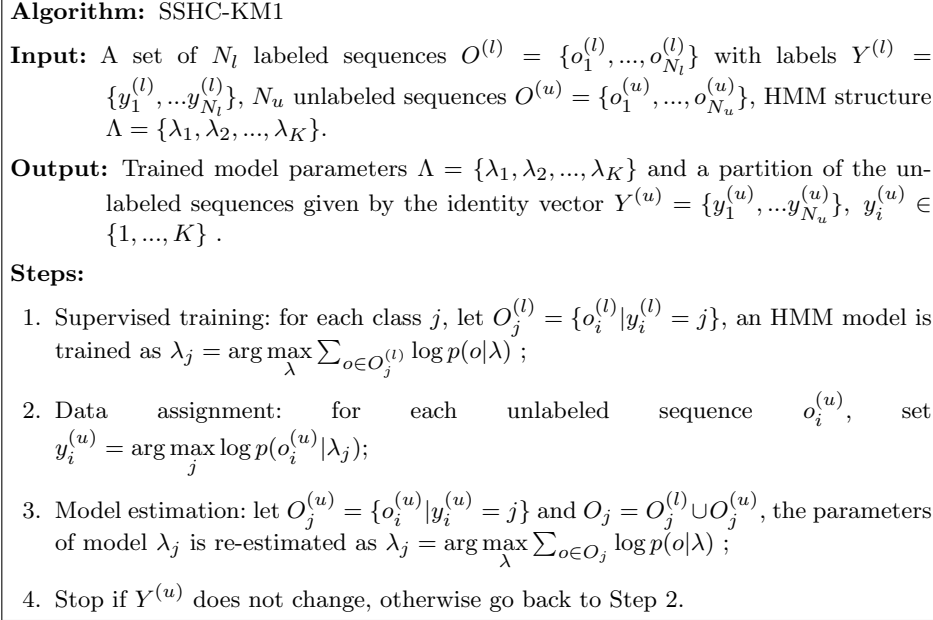


Fig. 2. Semi-supervised HMM-based classification - 1.

labeled sequences and the newly-labeled sequences. In this version, the labeled sequences are used to constrain the whole training process of HMMs.

We call **Step 1** of the SSHC-KM1 algorithm *supervised* step and **Step 2&3** *semi-supervised* steps. The other two versions of the SSHC algorithm differ from the first one in just the semi-supervised steps. Figure 3 shows only the **Step 2&3** for the SSHC-KM2 algorithm, which frees labeled sequences after the supervised step and consequently a labeled sequence may be assigned a different label in the later semi-supervised iterative process. This is based on the idea that there may be noise (e.g., mislabeled sequences in the labeled data) that may prevent the HMMs from fitting (and classifying) unlabeled sequences better.

For SSHC-KM3, shown in Figure 4 (again, only **Step 2&3**), we use the trained HMMs from supervised step as a starting point for clustering on unlabeled sequences. That is, we use only unlabeled sequences in the semi-supervised steps. Intuitively, this one should underperform SSHC-KM1 and SSHC-KM2 since it uses less information (unlabeled data only) after the supervised step. We include it here, however, for completeness. The words in boldface in Figure 3 & 4 highlight the key differences of these algorithms.

It can be easily verified that the complexity for SSHC-KM1 and SSHC-KM2 is $O(KMM_1NTN_h^2)$, where K is the number of clusters, N the number of sequences, M the number of semi-supervised iterations, M_1 the number of iterations used for maximum likelihood estimation of an HMM model, T the sequence length, and N_h

Algorithm: SSHC-KM2 (Step 2&3)

Steps:

2. Data assignment: for every (**labeled and unlabeled**) sequence o_i , set $y_i = \arg \max_j \log p(o_i | \lambda_j)$;
3. Model estimation: let $O_j = \{o_i^{(l)} | y_i^{(l)} = j\} \cup \{o_i^{(u)} | y_i^{(u)} = j\}$, the parameters of model λ_j is re-estimated as $\lambda_j = \max_{\lambda} \sum_{o \in O_j} \log p(o | \lambda)$;

Fig. 3. Semi-supervised HMM-based classification - 2.

Algorithm: SSHC-KM3 (Step 2&3)

Steps:

2. Data assignment: for each unlabeled sequence $o_i^{(u)}$, set $y_i^{(u)} = \arg \max_j \log p(o_i^{(u)} | \lambda_j)$;
3. Model estimation: let $O_j^{(u)} = \{o_i^{(u)} | y_i^{(u)} = j\}$, **using only unlabeled sequences**, the parameters of model λ_j is re-estimated as $\lambda_j = \max_{\lambda} \sum_{o \in O_j^{(u)}} \log p(o | \lambda)$;

Fig. 4. Semi-supervised HMM-based classification - 3.

the number of hidden states. Note the complexity of training a univariate HMM is $O(M_1NTN_h^2)$. SSHC-KM3 has slightly lower complexity since it uses only N_u unlabeled sequences (instead of all N sequences) in the semi-supervised steps.

5. Experimental Study

5.1. Datasets

We experiment on two datasets—a synthetic HMM-generated dataset (*syn200*) and a real EEG dataset. The synthetic dataset is an expanded version of the one used by [24]. 200 sequences of length $T(= 200)$ are generated from two continuous HMM models (HMM1 and HMM2), 100 from each. The number of hidden states is 2 for both models. The prior and observation parameters for HMM1 and HMM2 are the same. The prior is uniform and the observation distribution is univariate Gaussian with mean $\mu = 3$ and variance $\sigma^2 = 1$ for hidden state 1, and mean $\mu = 0$ and variance $\sigma^2 = 1$ for hidden state 2. The state transition parameters of HMM1 and HMM2 are $A_1 = \begin{bmatrix} 0.6 & 0.4 \\ 0.4 & 0.6 \end{bmatrix}$ and $A_2 = \begin{bmatrix} 0.4 & 0.6 \\ 0.6 & 0.4 \end{bmatrix}$, respectively. A sample sequence from each category is shown in Figure 5, along with a sequence of random numbers. One can see that the two HMM-generated sequences are different from the random numbers but not easy to be separated from each other.

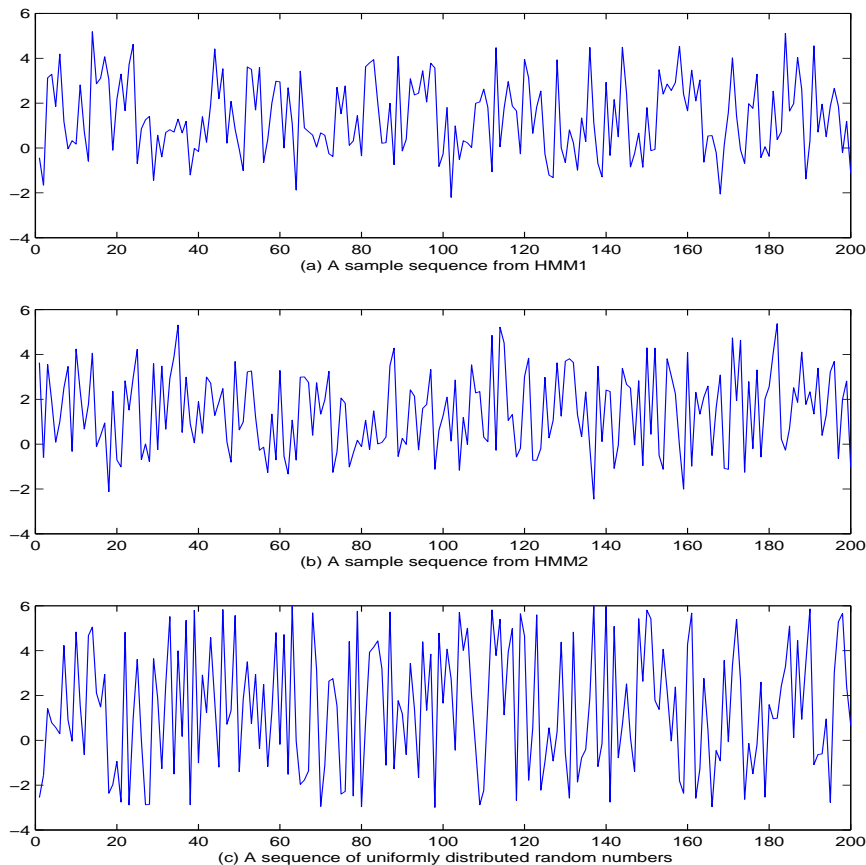
8 *Shi Zhong*

Fig. 5. Synthetic sequence examples.

The real dataset is a small EEG dataset downloaded from UCI KDD Archive web site (<http://kdd.ics.uci.edu/>). It contains measurements from 64 electrodes on a human scalp. In our experiment we only extract data from one electrode^a (as shown in Figure 6) and model it with a univariate HMM. Multiple electrodes can be modeled with multivariate HMMs but are not the focus of this paper. There are 20 measurements from two subjects, a control subject and an alcoholic subject, 10 from each. The measurement is sampled at 256Hz for 1 second, producing a sequence length of 256. The goal is to classify the subject as normal or alcoholic based on the EEG time-series data.

Geva and Kerem [8] clustered EEG time-series using a weighted fuzzy k-means algorithm on extracted feature vector space. But expert knowledge is required to extract good features. In this paper no feature extraction is needed; the raw time-

^aWe randomly picked one from several commonly studied channels in EEG signal analysis [10].

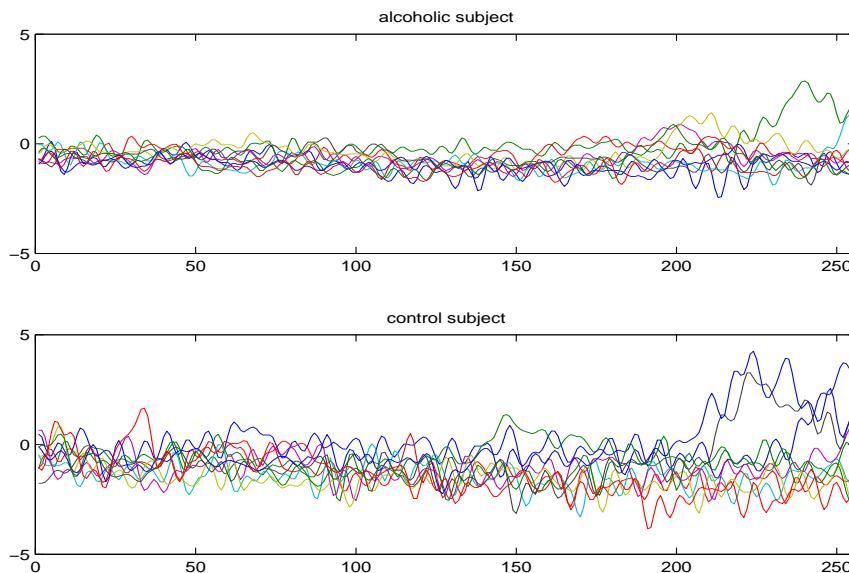


Fig. 6. EEG data samples for one alcoholic subject and one control subject.

series are modeled with HMMs. EEG signal is believed to be highly correlated with the sleep stages of brain cells. The number of sleep stages is about 5 or 6 according to [8]. Therefore, the correct number of hidden states is assumed to be 5 or 6.

5.2. Experimental Setting

A few details on the parameter estimation of HMMs are worth mentioning here. It is well-known that using mixture of Gaussians as the observation model of an HMM sometimes results in singularity problems during training. Juang et al. [13] suggested solving the problem by re-training from a different initialization. We chose to use maximum *a posteriori* (MAP) training [7] which is a more robust way of estimating HMM parameters. It has also been observed [21] that accurately estimating the means of Gaussians is essential to learning good models for continuous HMMs. We used the standard k-means algorithm to locate the means for the observation Gaussian distributions, following the approach used in [24]. Random initialization is used for other parameters.

In addition to the initialization scheme used above, we try to exploit the labeled information by rejecting random initializations that fail to generate models better than a baseline model (that is, the random guess model with 50% classification accuracy). Our experiences indicate that this is useful in improving classification accuracy by getting rid of some bad starting points.

For the EEG data, we scale all values (the value of every labeled and unlabeled sequence at every time slice) to be within $[-5, 5]$ to avoid severe mismatches between

data and initial random models.

For supervised classification, we need to specify part of the data as labeled data. To see how the number of labeled sequences affect the performance of semi-supervised learning, we experiment on different number of labeled sequences. We vary the number from 2 to 90 for the synthetic data and from 2 to 8 for the real EEG data. Given a specified N_l (number of labeled sequences), we randomly pick $N_l/2$ samples from class 1 and $N_l/2$ from class 2. For transductive learning, we evaluate the classification performance on all (designated) unlabeled sequences; for regular semi-supervised learning, we fix half sequences as an independent test set, on which the classification error is measured, and labeled sequences come only from the other half (training set).

We run each experiment 10 times and report the average classification errors. Statistical t -test results are computed based on the 10 runs, with a p -value of 0.05 as the significance threshold. We manually set the complexity of HMMs *a priori*, that is, the number of hidden states is not automatically selected. However, we tried different numbers of hidden states for each dataset, 2, 5, and 8 for the synthetic dataset and 5, 8, and 11 for the EEG dataset, to see the effect of model complexity.

5.3. Results Analysis

Figure 7 shows the classification results on the *syn200* dataset. On the left are results for regular semi-supervised learning and on the right are results for transductive learning. Note that the error rates for the two learning paradigms were measured on different unlabeled sets, thus are not exactly comparable. While the number of unlabeled test sequences is fixed at 100 for regular semi-supervised learning, it changes with number of labeled sequences for transductive learning. For example, when $N_l = 20$, the error rate is measured on the rest 180 unlabeled sequences for transductive learning. Nonetheless, we do see similar error ranges for the two learning paradigms given the same N_h value.

Each row shows results for a different model complexity (i.e., number of hidden states $N_h = 2, 5, \text{ and } 8$, respectively). In each figure, five curves of misclassification rate (y -axis) vs. number of labeled sequences (x -axis) are shown, corresponding to supervised method, semi-supervised EM (SSHC-EM2, an EM version of SSHC-KM2), and the three k-means based semi-supervised algorithms presented in this paper, respectively. The number of labeled sequences ranges from 2 up to 90. Figure 8 shows the same type of results on the *EEG-2* dataset, but with a different set of N_l 's (2 to 8) and N_h 's (5, 8, and 11, respectively).

First of all, a surprising result is the significant effect of N_h . Higher N_h values not only result in better error rates in all columns, but also are necessary for the superior performance of semi-supervised learning to be seen (especially for the synthetic dataset). Comparing the three rows of each column in both figures, one can see that increasing the complexity (number of hidden states) of HMM models leads to lower classification error, even though more hidden states intuitively lead to more

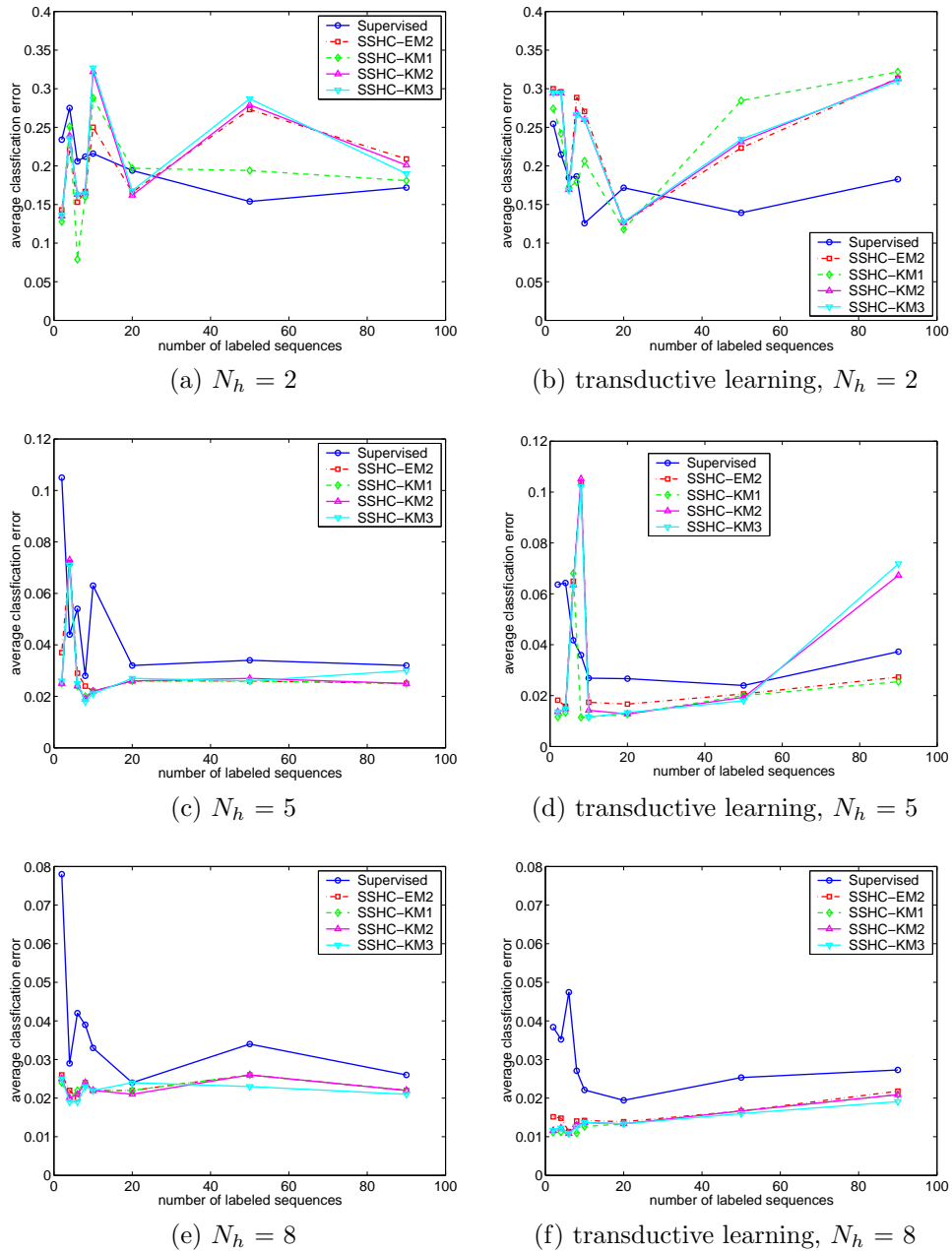


Fig. 7. Results on the *syn200* dataset.

local maxima and the numbers exceed “correct” values (2 for synthetic data and 5 or 6 for EEG data). We suspect the reason is that, since labeled sequences provides

12 Shi Zhong

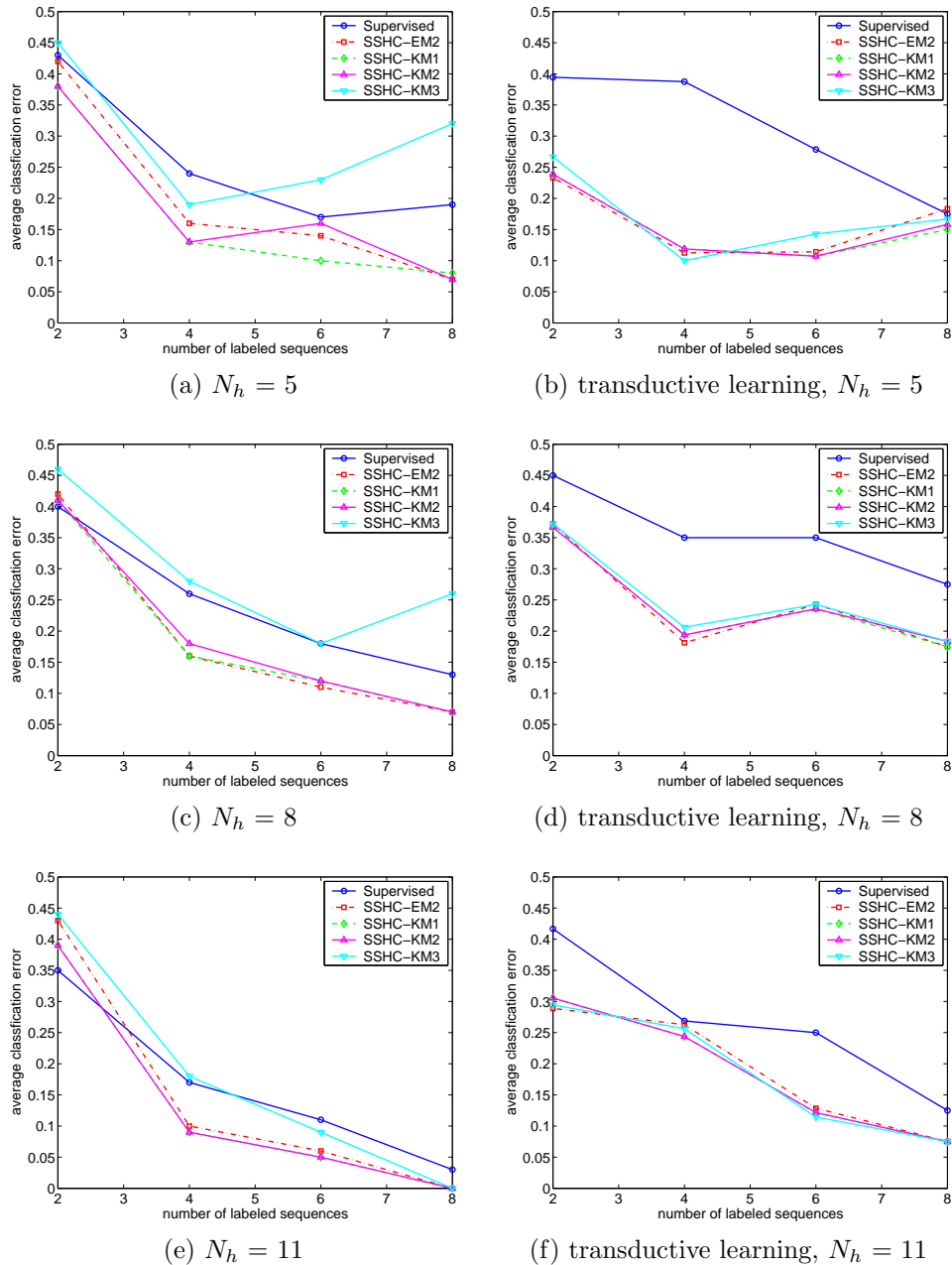


Fig. 8. Results on the *EEG-2* dataset.

a good starting point, the benefit of getting more discriminating power from more complex models outweigh the negative effect of having more local maxima.

Secondly, we can see that SSHC-KM1 and SSHC-KM2 perform very similarly in almost all situations (and when they do differ, SSHC-KM1 seems to be the better one). This makes sense since these two algorithms both use all data, i.e., see the same amount of information for semi-supervised training. It indicates that the noise contained in labeled data is not an important factor in most cases. The SSHC-KM3 algorithm, which uses only unlabeled sequences after the supervised step, fares worse than SSHC-KM1 and SSHC-KM2 in some cases, e.g., in Figure 8(a),(c), and (e) on the EEG dataset. Our t -test results (not shown for simplicity) indicate that SSHC-KM3 performs significantly worse in the aforementioned cases but never significantly outperforms SSHC-KM1 or SSHC-KM2. The results suggest that labeled sequences can contribute in the whole process—both supervised step and semi-supervised step. In all figures, results of the SSHC-EM2 algorithm are close to those of SSHC-KM1 and SSHC-KM2 algorithms, confirming that soft EM assignment and hard k-means assignment give similar results for learning HMMs.

Now let us look at SSHC-KM1 alone and examine the benefits of unlabeled sequences and the effect of N_l on semi-supervised results. The results clearly show that unlabeled sequences improve the classification accuracy significantly in most situations for the EEG dataset, and in some situations (when N_h is large, i.e., $N_h = 8$) for the synthetic dataset. Based on our t -test results, SSHC-KM1 is significantly better than the supervised algorithm in all but a few cases in Figure 7 (e)&(f) and Figure 8. The few cases are easy to identify, e.g., $N_l = 20$ in Figure 7 (e), $N_l = 6$ in Figure 8 (a), $N_l = 8$ in Figure 8 (b), and $N_l = 4$ in Figure 8 (f). The $N_l = 2$ cases for the left column of Figure 8 will be discussed later at the end of this section.

For the *syn200* dataset, the error rate of semi-supervised learning is relatively “flat” and there is no downward trend. The reason is that the synthetic dataset is relatively easy to classify and the performance of semi-supervised learning settled quickly even for small N_l values. For the EEG dataset, the downward trend of semi-supervised learning error rates is more evident as N_l grows.

Finally, let us examine the difference between transductive learning and regular semi-supervised learning. For the synthetic dataset, no trends can be seen for small N_h 's, but for $N_h = 8$, transductive learning produces lower error rates (especially for small N_l 's). For the EEG dataset (Figure 8), it is evident that when N_l is small (e.g., 2), transductive learning performs much better than supervised learning while regular semi-supervised learning does not or even does the opposite. The reason is that semi-supervised learning does not get enough guidance about classifying unlabeled test sequences when N_l is small. In transductive learning, however, the test sequences are used to help train HMMs, thus the final HMM models may be better tuned to separating different classes in the unlabeled test sequences, even for $N_l = 2$.

6. Related Work

While there is only limited amount of existing work on semi-supervised learning of HMMs, related work on general semi-supervised learning algorithms and applications is abundant. We describe some here as readers may find them interesting and they may lead to new algorithms for training HMMs using both labeled and unlabeled sequences.

Blum et al. [4] introduced co-training algorithm for learning from labeled and unlabeled data. They assume that there exist two independent sets of features and either set can confidently predict the class labels of unlabeled data. Their method performs well for real text data despite the strong assumption. Joachims [12] proposed transductive SVM—a method to incorporate unlabeled data into the formulation of a SVM classifier. The basic idea is to keep unlabeled data far away from the decision boundary in addition to trying to maximize the decision margin for labeled data. Good performance of the transductive SVM has been shown on text classification problems.

Guerrero-Curienes and Cid-Sueiro [9] proposed to minimize the cost function of a classifier using labeled data and minimize a corresponding entropy measure using unlabeled data. In their formulation, minimizing entropy is equivalent to minimizing uncertainty of unlabeled data, which has the same flavor of forcing unlabeled data to be away from the most uncertain region (i.e. decision boundary) as transductive SVM. Their method is applicable only when the classifier outputs class probabilities. Muslea et al. [16] combined semi-supervised methods discussed in [18] with active learning to generate robust multi-view learning algorithms. But the comparison between semi-supervised methods and their combined methods are not on the same labeled data, which undermines the significance of their results. For their methods, some of the labeled data is picked out by active learning strategy. Blum and Chawla [3] proposed to use graph mincut method to do semi-supervised learning. Their method performs comparably with other (e.g. EM) methods. A similarity measure between any two data instances is needed, and the computational effort to construct the graph seems to be high.

Our work is most similar to [11], where the EM algorithm is used to train HMMs with both labeled and unlabeled sequences. It has been shown that the EM approach can improve the classification performance substantially on some data while hurting accuracy on other ones [19]. The negative results are often attributed to severe mismatch between models and data.

There have been some studies on relative values of labeled and unlabeled data for classification. Unfortunately, results from these studies suggest little value with unlabeled data relative to labeled data. Castelli and Cover [5] proved that labeled examples are exponentially more valuable than unlabeled examples in pattern recognition tasks. But they make very strong assumptions that the input distribution is known completely and that all class-conditional distributions can be learned from unlabeled data only. These assumptions usually do not hold in reality. In a recent

study, Zhang and Oles [27] also questioned the usefulness of transductive SVMs.

Nigam and Ghani [18] compared co-training with EM algorithm and incremental vs. iterative approach on text classification problems. Incremental approach seems to work better. They argue that EM is not geared toward classification task while co-training generates more discriminative classifiers. So it is not surprising that co-training outperforms EM in their experiments.

Semi-supervised learning can be viewed from a different aspect: Labeled data provide a good starting point for clustering on unlabeled data. Wagstaff and Cardie [25] used some constraints type knowledge to help clustering. The constraints they used (must-link, cannot-link) can be seen as labeled data in a classification task.

Finally, Seeger [23] provided a good summary of recent developments in semi-supervised learning with labeled and unlabeled data.

7. Concluding Remarks

For the time-series classification problems studied in this paper, we have shown that unlabeled sequences can improve classification accuracy significantly when the model capacity (number of hidden states in HMMs) is reasonably high. We also observed in our experiments that more complex models achieve higher classification accuracy. Three different strategies of combining labeled and unlabeled data have been investigated for learning HMMs for sequence classification. The strategies that use labeled information throughout the model training process are the superior ones according to our experiments.

Two different semi-supervised learning paradigms have been discussed and experimentally studied. Transductive learning can be good when the number of labeled sequences is very small and when it is possible to use test data and training data together.

Future work can proceed in the following directions:

- Incrementally labeling the unlabeled data [18] may improve the semi-supervised learning results because of its less greedy behavior. Currently, all the unlabeled data are “labeled” (for model training) in batch style at every iteration.
- By integrating model selection method with semi-supervised learning, one may investigate the interaction between semi-supervised learning and model selection.

References

1. S. Basu, A. Banerjee, and R. Mooney. Semi-supervised clustering by seeding. In *Proc. 19th Int. Conf. Machine Learning*, pages 19–26, Sydney, Australia, July 2002.
2. L. E. Baum. An inequality and associated maximization technique in statistical estimation for probabilistic functions of Markov processes. *Inequalities*, 3:1–8, 1969.

16 REFERENCES

3. A. Blum and S. Chawla. Learning from labeled and unlabeled data using graph mincuts. In *Proc. 18th Int. Conf. Machine Learning*, pages 19–26, 2001.
4. A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *The 11th Annual Conf. Computational Learning Theory*, pages 92–100, 1998.
5. V. Castelli and T. M. Cover. The relative value of labeled and unlabeled samples in pattern recognition with an unknown mixing parameter. *IEEE Trans. Information Theory*, 42(6):2102–2117, November 1996.
6. A. Dong and B. Bhanu. A new semi-supervised em algorithm for image retrieval. In *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition*, volume 2, pages 662–667, Madison, MI, June 2003.
7. J.-L. Gauvain and C.-H. Lee. Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains. *IEEE Trans. Speech and Audio Processing*, 2(2):291–298, April 1994.
8. A. B. Geva and D. H. Kerem. Brain state identification and forecasting of acute pathology using unsupervised fuzzy clustering of EEG temporal patterns. In H.-N. Teodoroescu, A. Kandel, and L. C. Jain, editors, *Fuzzy and Neuro-Fuzzy Systems in Medicine*, chapter 3, pages 57–93. CRC Press, 1998.
9. A. Guerrero-Curieses and J. Cid-Sueiro. An entropy minimization principle for semi-supervised terrain classification. In *Proc. IEEE Int. Conf. Image Processing*, volume 3, pages 312–315, 2000.
10. E. Haselsteiner and G. Pfurtscheller. Using time-dependent neural networks for EEG classification. *IEEE Trans. Rehabilitation Engineering*, 8(4):457–463, December 2000.
11. M. Inoue and N. Ueda. Exploitation of unlabeled sequences in hidden markov models. *IEEE Trans. Pattern Anal. Machine Intell.*, 25(12):1570–1581, December 2003.
12. T. Joachims. Transductive inference for text classification using support vector machines. In *Proc. 16th Int. Conf. Machine Learning*, pages 200–209, 1999.
13. B.-H. Juang, S. E. Levinson, and M. M. Sondhi. Maximum likelihood estimation for multivariate mixture observations of Markov chains. *IEEE Trans. Information Theory*, 32(2):307–309, 1986.
14. M. Kearns, Y. Mansour, and A. Y. Ng. An information-theoretic analysis of hard and soft assignment methods for clustering. In *Proc. 13th Conf. Uncertainty in Artificial Intelligence*, pages 282–293, 1997.
15. C. Li and G. Biswas. Applying the hidden Markov model methodology for unsupervised learning of temporal data. *International Journal of Knowledge-based Intelligent Engineering Systems*, 6(3):152–160, July 2002.
16. I. Muslea, S. Minton, and C. A. Knoblock. Selective sampling + semi-supervised learning = robust multi-view learning. In *IJCAI workshop on Text Learning: Beyond Supervision*, Seattle, Washington, August 2001.
17. K. Nigam. *Using Unlabeled Data to Improve Text Classification*. PhD thesis, School of Computer Science, Carnegie Mellon University, May 2001.

18. K. Nigam and R. Ghani. Understanding the behavior of co-training. In *KDD Workshop on Text Mining*, 2000.
19. K. Nigam, A. McCallum, S. Thrun, and T. Mitchell. Text classification from labeled and unlabeled documents using EM. *Machine Learning*, 39(2/3):103–134, 2000.
20. L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of The IEEE*, 77(2):257–286, 1989.
21. L. R. Rabiner, B.-H. Juang, S. E. Levinson, and M. M. Sondhi. Some properties of continuous hidden Markov model representations. *AT&T Technical Journal*, 64(6):1251–1269, 1985.
22. I. Rezek and S. J. Roberts. Estimation of coupled hidden Markov models with application to biosignal interaction modeling. In *Proc. IEEE Int. Conf. Neural Network for Signal Processing*, volume 2, pages 804–813, 2000.
23. M. Seeger. Learning with labeled and unlabeled data. Technical report, Institute for Adaptive and Neural Computation, University of Edinburgh, February 2001.
24. P. Smyth. Clustering sequences with hidden Markov models. In M. C. Mozer, M. I. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems 9*, pages 648–654. MIT Press, 1997.
25. K. Wagstaff and C. Cardie. Clustering with instance-level constraints. In *Proc. 17th Int. Conf. Machine Learning*, pages 1103–1110, 2000.
26. Y. Wu and T. S. Huang. Self-supervised learning for visual tracking and recognition of human hand. In *Proc. 17th National Conference on Artificial Intelligence*, pages 243–248, July 2000.
27. T. Zhang and F. Oles. A probabilistic analysis on the value of unlabeled data for classification problems. In *Proc. 17th Int. Conf. Machine Learning*, pages 1191–1198, June 2000.
28. S. Zhong. Semi-supervised sequence classification with hmms. In *17th International FLAIRS Conference (FLAIRS 2004)*, pages 568–573, Miami Beach, FL, May 2004.
29. S. Zhong and J. Ghosh. HMMs and coupled HMMs for multi-channel EEG classification. In *Proc. IEEE Int. Joint Conf. Neural Networks*, pages 1154–1159, May 2002.
30. S. Zhong and J. Ghosh. A unified framework for model-based clustering. *Journal of Machine Learning Research*, 4:1001–1037, November 2003.

Biographical Sketch and Photo