Sequence Selection for Active Learning

Brigham Anderson Sajid Siddiqi Andrew Moore

CMU-RI-TR-06-16

April 2006

Robotics Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213

Abstract

Scarcity of labelled data often hampers the learning of Hidden Markov Models in applications such as speech, text, and video processing. Although current active learning algorithms can select examples to be labelled, it is not clear how to select examples to be completely annotated.

This work presents a novel information gain solution to the problem. The algorithm can either select the best sequence from a set of sequences, or extract the best subsequence from an unsegmented stream of data. By using dynamic programming, the computation can be performed in time linear in the number of timesteps. These results apply to any time series model having the Markov property.

Contents

1 Introduction

Active learning algorithms are used to reduce the number of labeled examples necessary for a given level of performance. The canonical active learning task consists of a model, an unlabelled dataset, and a labeller (such as a human expert.) The active learning algorithm greedily selects an "informative" example from the unlabelled dataset to present to the labeller. Then the newly-labelled example is added to the dataset, the model is relearned, and the process repeated.

The simplest approach to active learning is *Uncertainty Sampling* in which the most uncertain example is selected for labelling (Lewis & Catlett, 1994). Although this approach is intuitive and easy to implement, it is sensitive to noise. A more sound set of methods selects examples that are expected to most reduce the version space, the *Query by Committee* (QBC) methods (Seung et al., 1992). The *Information Gain* (IG) approach is similar to QBC, but directly attempts to minimize the entropy of the model posterior (Mackay, 1992; Anderson & Moore, 2005; Ji et al., 2006). Another option is to select examples that are expected to reduce classification error on a test set, the *error reduction* approach (Roy & McCallum, 2001).

With HMMs, there are several possible active learning tasks, the most common are listed here. Assume that one has a large set of unlabelled sentences:

1. Select a Sequence for Classification. E.g., choose a sentence to have labelled as either English or Not-English. Previous work includes (Ji et al., 2006; Tur et al., 2003)

2. Select a Token for Annotation. E.g., choose a single word to get the Part-of-Speech (POS) label of. Previous work includes (Scheffer et al., 2001; Anderson & Moore, 2005)

3. Select a Sequence for Annotation. E.g., choose one sentence to have every word POS labelled. Previous work includes (Dagan & Engelson, 1995)

4. Extract a Subsequence for Annotation. E.g., choose any contiguous subset of words to have POS labelled.

Information gain solutions to Tasks 1 and 2 were put forth in (Ji et al., 2006) and (Anderson & Moore, 2005) respectively. The present work provides information gain solutions to Tasks 3 and 4. Note that other possibilities for the "sentences" and "words" of the tasks could be genes and nucleotides or video clips and frames.

Section 2 contains an overview of the role of information gain in active learning. Section 4 reviews the information gain approach to the token-selection task. Section 5 introduces new measures for information gain of a sequence. Sections 6 and 7 explain how to efficiently extract the optimal subsequence using dynamic programming.

2 Active Learning with Information Gain

We have a set of observations $\mathbf{y}_{1:T} = \{y_1, y_2, ..., y_T\}$ which has a one-to-one correspondence with a hidden set of true classes $\mathbf{x}_{1:T} = \{x_1, x_2, ..., x_T\}$. Each example can be either observed or unobserved. There is a set of labels $\mathbf{l}_{1:T} = \{l_1, l_2, ..., l_T\}$. If the ith label is observed, then $l_i = x_i$, otherwise, $l_i = *$. The set of observations and labels we will denote by $\mathcal{D} = \{\mathbf{y}_{1:T}, \mathbf{l}_{1:T}\}$.

The random variables $\mathbf{X}_{1:T} = \{X_1, X_2, ..., X_T\}$ will be used to represent the learner's distribution over the labels. We also have a model that can estimate $P(X_i|y_i, \theta)$, where $\theta \in \Theta$ are the model parameters. We also have a learning procedure that determines a posterior over models, $P(\Theta|\mathcal{D})$ or hereafter $P_{\mathcal{D}}(\Theta)$.

For information gain, we wish to minimize the entropy of $P_{\mathcal{D}}(\Theta)$ by carefully choosing an X_i to obtain the label of (Mackay, 1992). What is the expected reduction in $H(\Theta)$ if we were to be told the label of X_i? By definition it is the mutual information $\mathbf{IG}(\Theta; X_i)$

$$\mathbf{IG}(\Theta; X_i) = H(\Theta) - H(\Theta|X_i) \tag{1}$$

where entropy is defined in the standard way as $H(\Theta) = \sum_{\theta \in \Theta} P_{\mathcal{D}}(\theta) \log P_{\mathcal{D}}(\theta)$ and conditional entropy is the expected value of $H(\Theta)$ given that we know the value of X_i.

The only difficulty here is that the space Θ may be continuous or otherwise too difficult to sum over. In QBC and other active learning approaches, the model space is approximated by sampling from $P_{\mathcal{D}}(\Theta)$. This sample will be called a "committee" and denoted by $\mathcal{C}$, where $\mathcal{C} = \{c_1, c_2, ..., c_{|\mathcal{C}|}\}$ and $c_i \sim P_{\mathcal{D}}(\Theta)$.

We will use the committee to approximate Equation 1, and then employ symmetry of mutual information to show an equivalence

$$\begin{aligned} \mathbf{IG}(\mathcal{C}; X_i) &= H(\mathcal{C}) - H(\mathcal{C}|X_i) &(2)\\ &= H(X_i) - H(X_i|\mathcal{C}) &(3) \end{aligned}$$

This simply states the powerful fact that the information gained about $\mathcal{C}$ by observing X_i is the same as the information gained about X_i by observing $\mathcal{C}$. The index of the best example is thus

$$\begin{aligned} i^* &= \underset{i \in 1..T}{\operatorname{argmax}} \, \mathbf{IG}(\mathcal{C}; X_i) \\ &= \underset{i \in 1..T}{\operatorname{argmax}} \, H(X_i) - H(X_i|\mathcal{C}) &(4) \end{aligned}$$

Note that Equation 3 has a straightforward interpretation. The *total* information of any kind to be had from the label X_t is exactly $H(X_t)$. However, some of that information is irrelevant to $P_{\mathcal{D}}(\mathcal{C})$, namely, the entropy that does not originate from uncertainty about the true model/classifier. The term $-H(X_i|\mathcal{C})$ removes this irrelevant information from the objective function.

As a bonus, Equation 3 sheds some light on the relationship between information gain, uncertainty sampling, and query by committee. Taking entropy as one's uncertainty measure, uncertainty sampling will choose the label with the maximum $H(X)$. This score function is equivalent to Equation 3 without the second term. However, in domains of noiseless data and deterministic classification, $H(X_i|\mathcal{C})$ is always zero, so their behavior will be equivalent.

In addition, if one also measures QBC "disagreement" over the label of X_i by $H(X_i)$ (equivalently described as the KL-Divergence from average beliefs (McCallum

& Nigam, 1998)), then its behavior in noiseless domains will also be identical to uncertainty sampling. Thus, the special case of noise-free examples and deterministic classifiers represents a convergence point for the three main active learning algorithms: uncertainty sampling, query by committee, and information gain.

3 HMM Active Learning

If the data is sequential and an HMM is the model used, active learning will take a slightly different form. The main difference being that the data can no longer be considered to be independent, and that the indices of the observations and labels now indicate an ordering of the data. We will switch from the term "examples" to "timesteps". All previous notation still applies, supplemented by the following definition of an HMM.

3.1 Hidden Markov Models (HMMs)

For ease of presentation, we will retrict discussion to discrete-output HMMs, but the results apply in the continuous case. A discrete-output HMM is defined by a state space, $\mathcal{X}$, an observation space $\mathcal{Y}$, and a parameter θ, which is a tuple of three parameters describing the transition probabilities, the observation probabilites, and the initial state probabilities.
$\mathcal{X}$: *state space*, a set of n states $\{1, \dots, n\}$.
$\mathcal{Y}$: *observation space*, a set of m symbols $\{1, \dots, m\}$.
$P(X_{t+1} = j | X_t = i)$ *transition probabilities* for $i, j \in \mathcal{X}$
$P(Y = j | X = i)$ *output probabilities* for $i \in \mathcal{X}$ and $j \in \mathcal{Y}$
$P(X_1 = i)$ *initial state probabilities* for $i \in \mathcal{X}$

The active learning algorithms presented in this work require only two standard HMM algorithms which are found in nearly all HMM implementations: EM model learning and Forward-Backward inference (Rabiner, 1990).

EM model learning for HMMs produces the maximum-likelihood model from labelled (or partially-labelled) data. The Forward-Backward algorithm performs inference given data $\mathcal{D}$ and HMM parameters. In $O(mn^2T)$, it produces several useful probabilities such as $P_\mathcal{D}(X_t)$ and $P_\mathcal{D}(X_t | X_{t-1})$ for $t \in 1..T$.

3.2 The HMM Committee

Recall that the methods from Section 2 require sampling from $P_\mathcal{D}(\Theta)$. How does one sample an HMM? As mentioned, the EM model learning algorithm only provides a single maximum likelihood model.

One intuitive, yet incorrect, method for generating a committee would be to do multiple EM learning runs from different starting points and use the different local-maximum HMMs to form the committee. Unfortunately, local maxima in model space have little to do with the posterior. If search gets trapped in a terrible local maximum for half of the restarts, assigning it equal posterior probability makes no sense.

In this work, we have assumed that each of the HMM's $n + m + 1$ multinomials was drawn from a separate independent Dirichlet distribution. The parameters of the

Dirichlets are determined from the number of "virtual" counts (of particular state-state transitions and state-observation events) as determined during the EM run. Given these Dirichlets, one can then simply sample from them to draw the multinomials for the HMM sample. Another approach to computing and sampling from $P_{\mathcal{D}}(\Theta)$ for HMMs uses Variational Bayes (Ji et al., 2006; MacKay, 1997).

4 Selecting a Timestep to Label

What is the information gain of one label in a sequence? In other words, if we have a sequence of observations $\mathbf{y}_{1:T}$ without labels, what is the expected reduction in $H(\mathcal{C})$ if we were told the label, X_t? Our best choice is still

$$\begin{aligned} t^* &= \operatorname*{argmax}_{t \in 1..T} \mathbf{IG}(\mathcal{C}; X_t) && (5) \\ &= \operatorname*{argmax}_{t \in 1..T} H(X_t) - H(X_t|\mathcal{C}) && (6) \end{aligned}$$

Computing the terms of which is straightforward once the Forward-Backward algorithm provides $P_{\mathcal{D}}(X_t|c)$ and $P_{\mathcal{D}}(X_t|X_{t-1}, c)$.

$$H(X_t) = \sum_i P_{\mathcal{D}}(X_t = i) \log P_{\mathcal{D}}(X_t = i) \tag{7}$$

The probability $P_{\mathcal{D}}(X_t = i)$ is actually marginalized over the committee members.

$$\begin{aligned} P_{\mathcal{D}}(X_t = i) &= \sum_{c \in \mathcal{C}} P_{\mathcal{D}}(c) P(X_t = i|c) && (8) \\ &= \frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} P(X_t = i|c) && (9) \end{aligned}$$

The probability $P_{\mathcal{D}}(c)$ is the probability of the cth committee member, which is $\frac{1}{|\mathcal{C}|}$ since the committee members have already been sampled from $P_{\mathcal{D}}(\Theta)$. The second term of Equation 6 is found by

$$H(X_t|\mathcal{C}) = \frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} \sum_{i}^{n} P_{\mathcal{D}}(X_t = i|c) \log P_{\mathcal{D}}(X_t = i|c) \tag{10}$$

The overall sequence of events for determining the most informative timestep to obtain the label of is thus:

1. Run the Forward-Backward algorithm on the data once for each committee member, yielding $P_{\mathcal{D}}(X_t|c)$ and $P_{\mathcal{D}}(X_t)$ for each timestep.

2. Compute the information gain $H(X_t) - H(X_t|\mathcal{C})$ for each timestep.

3. Select the timestep with the maximum information gain.

5 Selecting a Sequence to Annotate

What is the information gain of a *sequence* of labels? In other words, we have a population of R observation sequences, the rth sequence of which is $\mathbf{y}_{1:T}^{(r)}$. Assume without loss of generality that all sequences are of length T. We can select one sequence for our labeller to completely annotate. So at each iteration of active learning we receive T labels.

The sequence which, when labelled, is expected to most reduce $H(\mathcal{C})$ will be

$$r^* = \underset{r \in 1..R}{\operatorname{argmax}} \, \mathbf{IG}(\mathcal{C}; \mathbf{X}_{1:T}^{(r)}) \tag{11}$$

$$= \underset{r \in 1..R}{\operatorname{argmax}} \, H(\mathbf{X}_{1:T}^{(r)}) - H(\mathbf{X}_{1:T}^{(r)} | \mathcal{C}) \tag{12}$$

So what is the entropy of a sequence of variables? We must compute the probability of every possible configuration, but the size of the space of possible labellings, $|\mathbf{X}_{1:T}|$, is m^T. This is prohibitively large. So, let us rewrite the equation according to the chain rule of entropy

$$\begin{aligned} H(\mathbf{X}_{1:T}) =& H(X_1) + H(X_2|X_1) + H(X_3|X_1, X_2) + ... \\ &+ H(X_T|X_1, X_2, X_3, ..., X_{T-1}) \end{aligned} \tag{13}$$

which doesn't solve the problem, but does allow us to apply the independencies of an HMM. In other words, since X_t is conditionally independent of X_{t-2} given X_{t-1}, we can say the following

$$H(\mathbf{X}_{1:T}) = H(X_1) + H(X_2|X_1) + H(X_3|X_2) + H(X_T|X_{T-1}) \tag{14}$$

Now we can rewrite the mutual information between a sequence of hidden labels and $\mathcal{C}$ as

$$\begin{aligned} \mathbf{IG}(\mathcal{C}; \mathbf{X}_{1:T}) =& H(X_1) - H(X_1|\mathcal{C}) \\ &+ \sum_{i=2}^{T} H(X_i|X_{i-1}) - H(X_i|X_{i-1}, \mathcal{C}) \end{aligned} \tag{15}$$

The first and second terms we have already seen, and the rest are easily derived from the Forward-Backward probabilities $P_{\mathcal{D}}(X_t, X_{t-1})$ (see appendix.) Equivalently,

$$\mathbf{IG}(\mathcal{C}; \mathbf{X}_{1:T}) = \mathbf{IG}(\mathcal{C}; X_1) + \sum_{i=2}^{T} \mathbf{IG}(\mathcal{C}; X_i | X_{i-1}) \tag{16}$$

So $\mathbf{IG}(\mathcal{C}; \mathbf{X}_{1:T})$ can be computed rather cheaply. The total cost to evaluate every sequence in our dataset is $O(R|\mathcal{C}|mn^2T)$, equivalent to running inference on each sequence $|\mathcal{C}|$ times.

It is clear from (15) that longer sequences will have an advantage. This behavior is understandable, since they will of course tend to contain more information. However, annotating longer sequences also requires more resources. One obvious solution is to assign a cost-per-label, γ, and penalize sequences according to their length. This term will play an important role in Section 7.

6 Selecting a Length-k Subsequence to Annotate

Suppose one is given a single long sequence of unlabelled observations of length T, and we are allowed to select any *subsequence* of length $k << T$ to be labelled. This is similar to the previous task, but instead of evaluating R separate sequences, we must now evaluate each possible length-k subsequence. Note that the token selection task from Section 4 is a special case of this task in which $k = 1$.

We will again use the information gain criterion to find the most informative subsequence of length k. The best interval, $\{t^*, t^* + k - 1\}$ will start at t^*

$$t^* = \underset{t \in 1..T-k+1}{\text{argmax}} \ \mathbf{IG}(\mathcal{C}; \mathbf{X}_{t:t+k-1}) \tag{17}$$

It will come as no surprise that this can be computed efficiently. First, once again compute $H(X_t)$, $H(X_t|X_{t-1})$, $H(X_t|\mathcal{C})$, and $H(X_t|X_{t-1}, \mathcal{C})$ for each timestep $t \in 1..T$. (see appendix.)

$$\begin{aligned} \mathbf{IG}(\mathcal{C}; \mathbf{X}_{t:t+k-1}) =& H(X_t) - H(X_t|\mathcal{C}) \\ &+ \sum_{i=t+1}^{t+k-1} H(X_i|X_{i-1}) - H(X_i|X_{i-1}, \mathcal{C}) \end{aligned} \tag{18}$$

Finding t^* then only requires evaluating Equation 18 for the intervals $[1, k]$, $[2, k + 1]$, ..., $[T - k, T]$. Each successive score can be computed efficiently from the previous score via a small number of additions and subtractions. The update equation is omitted but can be easily obtained by inspection of Equation 18. Once again, the computational cost is $O(|\mathcal{C}|mn^2T)$, which is independent of k.

7 Selecting an Any-Length Subsequence to Annotate

Suppose one is given a long sequence of unlabelled observations of length T, and are allowed to select *any subsequence* to be completely annotated. Of course, the most informative subsequence will always be the entire sequence, so one must specify a cost-per-label, γ, which will specify the minimum bits per label that must be expected from the best sequence.

With the inclusion of the cost term, the optimal interval is thus

$$\{t^*, w^*\} = \underset{t \in 1..T, w \in t..T}{\text{argmax}} \ \mathbf{IG}(\mathcal{C}; \mathbf{X}_{t:w}) - \gamma \cdot (w - t) \tag{19}$$

The naive approach to solving (19) would evaluate every possible interval, which is quadratic in T. Fortunately, a dynamic programming solution exists. Define

$$\alpha_t = H(X_t) - H(X_t|\mathcal{C}) - \gamma \tag{20}$$

$$\beta_t = H(X_t|X_{t-1}) - H(X_t|X_{t-1}, \mathcal{C}) - \gamma \tag{21}$$

so that the problem can be written as

$$\{t^*, w^*\} = \underset{t \in 1..T, w \in t..T}{\text{argmax}} \ \alpha_t + \sum_{i=t+1}^{w} \beta_i \tag{22}$$

The algorithm for finding $\{t^*, w^*\}$ scans through the sequence once while maintaining an internal state which contains the best interval seen so far and the beginning of the best interval that ends at the current timestep. The cost is once again $O(|\mathcal{C}|mn^2T)$, no more expensive than $|\mathcal{C}|$ Forward-Backward iterations.

```
Define:
$a(t)$ : start of best interval in $1...t$
$b(t)$ : end of best interval in $1...t$
$c(t)$ : start of best interval which ends at $t$
score$^*(t)$ : $\alpha_{a(t)} + \sum_{i=a(t)+1}^{b(t)} \beta_i$
score$(t)$ : $\alpha_{c(t)} + \sum_{i=c(t)+1}^{t} \beta_i$


Initialization:
$t = 1$
$a(1) = b(1) = c(1) = 1$
score$^*(1) = $ score$(1) = 0$


Loop:
$t \leftarrow t + 1$

if (score$(t-1) + \beta_t > \alpha_t$)
then
    $c(t) = t$
    score$(t) = \alpha_t$
else
    $c(t) = c(t-1)$
    score$(t) = $ score$(t-1) + \beta_t$

if (score$(t) >$ score$^*(t)$)
then
    $a(t) = c(t)$
    $b(t) = t$
    score$^*(t) = $ score$(t)$
else
    $a(t) = a(t-1)$
    $b(t) = b(t-1)$
    score$^*(t) = $ score$^*(t-1)$


Termination:
if ($t == T$)
then
    return $a(T), b(T)$
```

Practical implementations of this algorithm would probably include a maximum-size parameter that limited the size of the subsequence that could be returned to be labelled.

8 Related Work

Dagan and Engelson (Dagan & Engelson, 1995) applied the QBC algorithm to selecting sentences to annotate with Part-of-Speech (POS) labels, the goal being to learn an HMM POS labeller. Sampling from the model posterior was achieved by assuming each parameter of each multinomial was an independent univariate truncated Normal distribution. The variance of these distributions were subsequently adjusted by a temperature parameter.

Disagreement between committee members in (Dagan & Engelson, 1995) was measured by "vote entropy" per timestep, which used each members most likely classification for a particular timestep. The average vote entropy was used to weight the probability of selecting a sequence. The vote entropy used was quite different from the entropy used in this work; it was based on classification disagreement, not the entropy of the label's actual posterior.

The selection of texts for complete annotation was also pursued in (Thompson et al., 1999), where the two tasks were semantic parsing and information extraction. The models were rule-based, not HMMs, and the active learning method used was uncertainty sampling.

9 Conclusions

This work described a novel active learning algorithm that selects maximally informative sequences for the purpose of actively learning HMMs. Several variants of the task were described, from selecting from a pre-segmented population of unlabelled sequences, to determining the optimal subsequence automatically from a single stream of data. All of the algorithms are equally inexpensive; they are linear in the total number of timesteps.

10 APPENDIX

During HMM inference, the vectors γ and the matrices ξ_t summarize several useful sets of probabilities (Rabiner, 1990).

$$\gamma_t[i] = P_{\mathcal{D}}(X_t = i) \tag{23}$$

$$\xi_t[i,j] = P_{\mathcal{D}}(X_t = i, X_{t-1} = j) \tag{24}$$

The quantities in Equations 23 and 24 are obtained in time $O(mn^2T)$ from the Forward-Backward algorithm. The computation of the entropies of a single hidden state in a sequence of hidden states is found as follows

$$H(X_t) = \sum_i P_{\mathcal{D}}(X_t = i) \log P_{\mathcal{D}}(X_t = i) \tag{25}$$

$$= \sum_i \hat{\gamma}_t[i] \log \hat{\gamma}_t[i] \tag{26}$$

where $\hat{\gamma}_t[i]$ is the state posterior $P_{\mathcal{D}}(X_t = i)$ marginalized over all models in the committee

$$\hat{\gamma}_t[i] = \frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} \gamma_t^c[i] \tag{27}$$

and $\gamma_t^c[i]$ is the state posterior given model c, or $P_{\mathcal{D}}(X_t = i|c)$. The conditional entropy of a label given $\mathcal{C}$ is

$$H(X_t|\mathcal{C}) = \frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} \sum_i^n P_{\mathcal{D}}(X_t = i|c) \log P_{\mathcal{D}}(X_t = i|c) \tag{28}$$

$$= \frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} \sum_i^n \gamma_t^c[i] \log \gamma_t^c[i] \tag{29}$$

and the conditional entropy of a state at time t given the previous state is

$$H(X_t|X_{t-1}) = \sum_i^n \hat{\gamma}_{t-1}[i] \sum_j^n \frac{\hat{\xi}_t[i,j]}{\hat{\gamma}_{t-1}[i]} \log \frac{\hat{\xi}_t[i,j]}{\hat{\gamma}_{t-1}[i]} \tag{30}$$

When the previous conditional entropy is also conditioned on $\mathcal{C}$,

$$H(X_t|X_{t-1}, \mathcal{C}) \tag{31}$$

$$= \frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} \sum_i^n P_{\mathcal{D}}(X_{t-1} = i|c) H(X_t|X_{t-1} = i, c) \tag{32}$$

$$= \frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} \sum_i^n \gamma_{t-1}^c[i] \sum_j^n \frac{\xi_t^c[i,j]}{\gamma_{t-1}^c[i]} \log \frac{\xi_t^c[i,j]}{\gamma_{t-1}^c[i]} \tag{33}$$

References

Anderson, B., & Moore, A. (2005). Active learning for hidden Markov models: Objective functions and algorithms. *Proceedings of ICML-05, International Conference on Machine Learning*. Morgan Kaufmann Publishers, San Francisco, US.

Dagan, I., & Engelson, S. P. (1995). Committee-based sampling for training probabilistic classifiers. *International Conference on Machine Learning* (pp. 150–157).

Ji, S., Krishnapuram, B., & Carin, L. (2006). Variational bayes for continuous hidden markov models and its application to active learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence (to appear)*.

Lewis, D. D., & Catlett, J. (1994). Heterogeneous uncertainty sampling for supervised learning. *Proceedings of ICML-94, 11th International Conference on Machine Learning* (pp. 148–156). New Brunswick, US: Morgan Kaufmann Publishers, San Francisco, US.

Mackay, D. (1992). Information-Based Objective Functions for Active Data Selection. *Neural Computation*, *4*, 589–603.

MacKay, D. (1997). Ensemble learning for hidden markov models.

McCallum, A. K., & Nigam, K. (1998). Employing EM in pool-based active learning for text classification. *Proceedings of ICML-98, 15th International Conference on Machine Learning* (pp. 350–358). Madison, US: Morgan Kaufmann Publishers, San Francisco, US.

Rabiner, L. R. (1990). A tutorial on hidden markov models and selected apllications in speech recognition. In A. Waibel and K.-F. Lee (Eds.), *Readings in speech recognition*, 267–296. San Mateo, CA: Kaufmann.

Roy, N., & McCallum, A. (2001). Toward optimal active learning through sampling estimation of error reduction. *Proc. 18th International Conf. on Machine Learning* (pp. 441–448). Morgan Kaufmann, San Francisco, CA.

Scheffer, T., Decomain, C., & Wrobel, S. (2001). Active hidden Markov models for information extraction. *Lecture Notes in Computer Science*, *2189*, 309+.

Seung, H. S., Opper, M., & Sompolinsky, H. (1992). Query by committee. *Computational Learning Theory* (pp. 287–294).

Thompson, C. A., Califf, M. E., & Mooney, R. J. (1999). Active learning for natural language parsing and information extraction. *Proc. 16th International Conf. on Machine Learning* (pp. 406–414). Morgan Kaufmann, San Francisco, CA.

Tur, G., Schapire, R., & Hakkani-Tur, D. (2003). Active learning for spoken language understanding.