Appears in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*. ACL Press, 2008.

An Analysis of Active Learning Strategies for Sequence Labeling Tasks

Burr Settles[*†]
[*]Dept. of Computer Sciences
University of Wisconsin
Madison, WI 53706, USA
bsettles@cs.wisc.edu

Mark Craven[†*]
[†]Dept. of Biostatistics & Medical Informatics
University of Wisconsin
Madison, WI 53706, USA
craven@biostat.wisc.edu

Abstract

Active learning is well-suited to many problems in natural language processing, where unlabeled data may be abundant but annotation is slow and expensive. This paper aims to shed light on the best active learning approaches for sequence labeling tasks such as information extraction and document segmentation. We survey previously used query selection strategies for sequence models, and propose several novel algorithms to address their shortcomings. We also conduct a large-scale empirical comparison using multiple corpora, which demonstrates that our proposed methods advance the state of the art.

1 Introduction

Traditional supervised learning algorithms use whatever labeled data is provided to induce a model. By contrast, *active learning* gives the learner a degree of control by allowing it to select which instances are labeled and added to the training set. A typical active learner begins with a small labeled set $\mathcal{L}$, selects one or more informative *query* instances from a large unlabeled pool $\mathcal{U}$, learns from these labeled queries (which are then added to $\mathcal{L}$), and repeats. In this way, the learner aims to achieve high accuracy with as little labeling effort as possible. Thus, active learning can be valuable in domains where unlabeled data are readily available, but obtaining training labels is expensive.

Such is the case with many *sequence labeling* tasks in natural language domains. For example, part-of-speech tagging (Seung et al., 1992; Lafferty et al., 2001), information extraction (Scheffer et al., 2001; Sang and DeMeulder, 2003; Kim et al., 2004), and document segmentation (Carvalho and Cohen, 2004) are all typically treated as sequence labeling problems. The source data for these tasks (i.e., text documents in electronic form) are often easily obtained. However, due to the nature of sequence labeling tasks, annotating these texts can be rather tedious and time-consuming, making active learning an attractive technique.

While there has been much work on active learning for classification (Cohn et al., 1994; McCallum and Nigam, 1998; Zhang and Oles, 2000; Zhu et al., 2003), active learning for sequence labeling has received considerably less attention. A few methods have been proposed, based mostly on the conventions of *uncertainty sampling*, where the learner queries the instance about which it has the least certainty (Scheffer et al., 2001; Culotta and McCallum, 2005; Kim et al., 2006), or *query-by-committee*, where a "committee" of models selects the instance about which its members most disagree (Dagan and Engelson, 1995). We provide more detail on these and the new strategies we propose in Section 3.

The comparative effectiveness of these approaches, however, has not been studied. Furthermore, it has been suggested that uncertainty sampling and query-by-committee fail on occasion (Roy and McCallum, 2001; Zhu et al., 2003) by querying outliers, e.g., instances considered informative in isolation by the learner, but containing little information about the *rest* of the distribution of instances. Proposed methods for dealing with these shortcomings have so far only considered classification tasks.

This paper presents two major contributions for active learning and sequence labeling tasks. First, we motivate and introduce several new query strategies for probabilistic sequence models. Second, we conduct a thorough empirical analysis of previously proposed methods with our algorithms on a variety of benchmark corpora. The remainder of this paper is organized as follows. Section 2 provides a brief introduction to sequence labeling and conditional random fields (the sequence model used in our experiments). Section 3 describes in detail all the query selection strategies we consider. Section 4 presents the results of our empirical study. Section 5 concludes with a summary of our findings.

2 Sequence Labeling and CRFs

In this paper, we are concerned with active learning for sequence labeling. Figure 1 illustrates how, for example, an information extraction problem can be viewed as a sequence labeling task. Let $\mathbf{x} = \langle x_1, \ldots, x_T \rangle$ be an observation sequence of length T with a corresponding label sequence $\mathbf{y} = \langle y_1, \ldots, y_T \rangle$. Words in a sentence correspond to tokens in the input sequence $\mathbf{x}$, which are mapped to labels in $\mathbf{y}$. These labels indicate whether the word belongs to a particular entity class of interest (in this case, org and loc) or not (null). These labels can be assigned by a sequence model based on a finite state machine, such as the one shown to the right in Figure 1.

We focus our discussion of active learning for sequence labeling on *conditional random fields*, or CRFs (Lafferty et al., 2001). The rest of this section serves as a brief introduction. CRFs are statistical graphical models which have demonstrated state-of-the-art accuracy on virtually all of the sequence labeling tasks mentioned in Section 1. We use linear-chain CRFs, which correspond to conditionally trained probabilistic finite state machines.

A linear-chain CRF model with parameters θ defines the posterior probability of $\mathbf{y}$ given $\mathbf{x}$ to be[1]:

$$P(\mathbf{y}|\mathbf{x};\theta) = \frac{1}{Z(\mathbf{x})} \exp\left(\sum_{t=1}^{T}\sum_{k=1}^{K} \theta_k f_k(y_{t-1}, y_t, \mathbf{x}_t)\right). \quad (1)$$

[1]Our discussion assumes, without loss of generality, that each label is uniquely represented by one state, thus each label sequence $\mathbf{y}$ corresponds to exactly one path through the model.

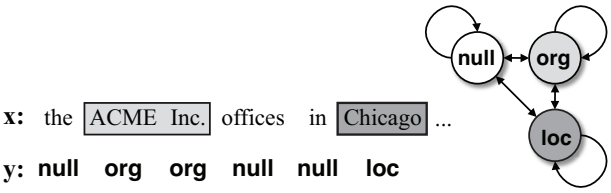

Figure 1: An information extraction example treated as a sequence labeling task. Also shown is a corresponding sequence model represented as a finite state machine.

Here $Z(\mathbf{x})$ is a normalization factor over all possible labelings of $\mathbf{x}$, and θ_k is one of K model parameter weights corresponding to some feature $f_k(y_{t-1}, y_t, \mathbf{x}_t)$. Each feature f_k describes the sequence $\mathbf{x}$ at position t with label y_t, observed along a transition from label states y_{t-1} to y_t in the finite state machine. Consider the example text from Figure 1. Here, f_k might be the feature WORD=*ACME* and have the value $f_k = 1$ along a transition from the null state to the org state (and 0 elsewhere). Other features set to 1 here might be ALLCAPS and NEXTWORD=*Inc.* The weights in θ are set to maximize the conditional log likelihood ℓ of training sequences in the labeled data set $\mathcal{L}$:

$$\ell(\mathcal{L};\theta) = \sum_{l=1}^{L} \log P(\mathbf{y}^{(l)}|\mathbf{x}^{(l)};\theta) - \sum_{k=1}^{K} \frac{\theta_k^2}{2\sigma^2},$$

where L is the size of the labeled set $\mathcal{L}$, and the second term is a Gaussian regularization penalty on $\|\theta\|$ to prevent over-fitting. After training, labels can be predicted for new sequences using the Viterbi algorithm. For more details on CRFs and their training procedures, see Sutton and McCallum (2006).

Note that, while we describe the active learning algorithms in the next section in terms of linear-chain CRFs, they have analogs for other kinds of sequence models, such as hidden Markov models, or HMMs (Rabiner, 1989), probabilistic context-free grammars (Lari and Young, 1990), and general CRFs (Sutton and McCallum, 2006).

3 Active Learning with Sequence Models

In order to select queries, an active learner must have a way of assessing how *informative* each instance is. Let $\mathbf{x}^*$ be the most informative instance according to some query strategy $\phi(\mathbf{x})$, which is a function used to evaluate each instance $\mathbf{x}$ in the unlabeled pool $\mathcal{U}$.

Given: Labeled set $\mathcal{L}$, unlabeled pool $\mathcal{U}$, query strategy $\phi(\cdot)$, query batch size B

repeat
  // learn a model using the current $\mathcal{L}$
  $\theta = \text{train}(\mathcal{L})$;
  for $b = 1$ **to** B **do**
    // query the most informative instance
    $\mathbf{x}_b^* = \arg\max_{\mathbf{x} \in \mathcal{U}} \phi(\mathbf{x})$;
    // move the labeled query from $\mathcal{U}$ *to* $\mathcal{L}$
    $\mathcal{L} = \mathcal{L} \cup \langle \mathbf{x}_b^*, \text{label}(\mathbf{x}_b^*) \rangle$;
    $\mathcal{U} = \mathcal{U} - \mathbf{x}_b^*$;
  end
until *some stopping criterion* ;

Algorithm 1: Pool-based active learning.

Algorithm 1 provides a sketch of the generic pool-based active learning scenario.

In the remainder of this section, we describe various query strategy formulations of $\phi(\cdot)$ that have been used for active learning with sequence models. We also point out where we think these approaches may be flawed, and propose several novel query strategies to address these issues.

3.1 Uncertainty Sampling

One of the most common general frameworks for measuring informativeness is *uncertainty sampling* (Lewis and Catlett, 1994), where a learner queries the instance that it is most uncertain how to label. Culotta and McCallum (2005) employ a simple uncertainty-based strategy for sequence models called **least confidence (LC)**:

$$\phi^{LC}(\mathbf{x}) = 1 - P(\mathbf{y}^*|\mathbf{x}; \theta).$$

Here, $\mathbf{y}^*$ is the most likely label sequence, i.e., the Viterbi parse. This approach queries the instance for which the current model has the least confidence in its most likely labeling. For CRFs, this confidence can be calculated using the posterior probability given by Equation (1).

Scheffer et al. (2001) propose another uncertainty strategy, which queries the instance with the smallest margin between the posteriors for its two most likely labelings. We call this approach **margin (M)**:

$$\phi^{M}(\mathbf{x}) = -\big(P(\mathbf{y}_1^*|\mathbf{x}; \theta) - P(\mathbf{y}_2^*|\mathbf{x}; \theta)\big).$$

Here, $\mathbf{y}_1^*$ and $\mathbf{y}_2^*$ are the first and second best label sequences, respectively. These can be efficiently computed using the N-best algorithm (Schwartz and Chow, 1990), a beam-search generalization of Viterbi, with $N = 2$. The minus sign in front is simply to ensure that ϕ^M acts as a maximizer for use with Algorithm 1.

Another uncertainty-based measure of informativeness is *entropy* (Shannon, 1948). For a discrete random variable Y, the entropy is given by $H(Y) = -\sum_i P(y_i) \log P(y_i)$, and represents the information needed to "encode" the distribution of outcomes for Y. As such, is it often thought of as a measure of uncertainty in machine learning. In active learning, we wish to use the entropy of our model's posteriors over its labelings. One way this has been done with probabilistic sequence models is by computing what we call **token entropy (TE)**:

$$\phi^{TE}(\mathbf{x}) = -\frac{1}{T} \sum_{t=1}^{T} \sum_{m=1}^{M} P_\theta(y_t = m) \log P_\theta(y_t = m), \tag{2}$$

where T is the length of $\mathbf{x}$, m ranges over all possible token labels, and $P_\theta(y_t = m)$ is shorthand for the marginal probability that m is the label at position t in the sequence, according to the model. For CRFs and HMMs, these marginals can be efficiently computed using the *forward* and *backward* algorithms (Rabiner, 1989). The summed token entropies have typically been normalized by sequence length T, to avoid simply querying longer sequences (Baldridge and Osborne, 2004; Hwa, 2004). However, we argue that querying long sequences should not be explicitly discouraged, if in fact they contain more information. Thus, we also propose the **total token entropy (TTE)** measure:

$$\phi^{TTE}(\mathbf{x}) = T \times \phi^{TE}(\mathbf{x}).$$

For most sequence labeling tasks, however, it is more appropriate to consider the entropy of the label sequence $\mathbf{y}$ as a whole, rather than some aggregate of individual token entropies. Thus an alternate query strategy is **sequence entropy (SE)**:

$$\phi^{SE}(\mathbf{x}) = -\sum_{\hat{\mathbf{y}}} P(\hat{\mathbf{y}}|\mathbf{x}; \theta) \log P(\hat{\mathbf{y}}|\mathbf{x}; \theta), \tag{3}$$

where $\hat{\mathbf{y}}$ ranges over all possible label sequences for input sequence $\mathbf{x}$. Note, however, that the number

of possible labelings grows exponentially with the length of $\mathbf{x}$. To make this feasible, previous work (Kim et al., 2006) has employed an approximation we call ***N*-best sequence entropy (NSE)**:

$$\phi^{NSE}(\mathbf{x}) = -\sum_{\hat{\mathbf{y}} \in \mathcal{N}} P(\hat{\mathbf{y}}|\mathbf{x};\theta) \log P(\hat{\mathbf{y}}|\mathbf{x};\theta),$$

where $\mathcal{N} = \{\mathbf{y}_1^*, \ldots, \mathbf{y}_N^*\}$, the set of the N most likely parses, and the posteriors are re-normalized (i.e., $Z(\mathbf{x})$ in Equation (1) only ranges over $\mathcal{N}$). For $N = 2$, this approximation is equivalent to ϕ^M, thus N-best sequence entropy can be thought of as a generalization of the margin approach.

Recently, an efficient entropy calculation via dynamic programming was proposed for CRFs in the context of semi-supervised learning (Mann and McCallum, 2007). We use this algorithm to compute the true sequence entropy (3) for active learning in a constant-time factor of Viterbi's complexity. Hwa (2004) employed a similar approach for active learning with probabilistic context-free grammars.

3.2 Query-By-Committee

Another general active learning framework is the *query-by-committee* (QBC) approach (Seung et al., 1992). In this setting, we use a committee of models $\mathcal{C} = \{\theta^{(1)}, \ldots, \theta^{(C)}\}$ to represent C different hypotheses that are consistent with the labeled set $\mathcal{L}$. The most informative query, then, is the instance over which the committee is in most disagreement about how to label.

In particular, we use the *query-by-bagging* approach (Abe and Mamitsuka, 1998) to learn a committee of CRFs. In each round of active learning, $\mathcal{L}$ is sampled (with replacement) L times to create a unique, modified labeled set $\mathcal{L}^{(c)}$. Each model $\theta^{(c)} \in \mathcal{C}$ is then trained using its own corresponding labeled set $\mathcal{L}^{(c)}$. To measure disagreement among committee members, we consider two alternatives.

Dagan and Engelson (1995) introduced QBC with HMMs for part-of-speech tagging using a measure called **vote entropy (VE)**:

$$\phi^{VE}(\mathbf{x}) = -\frac{1}{T}\sum_{t=1}^{T}\sum_{m=1}^{M} \frac{V(y_t, m)}{C} \log \frac{V(y_t, m)}{C},$$

where $V(y_t, m)$ is the number of "votes" label m receives from all the committee member's Viterbi labelings at sequence position t.

McCallum and Nigam (1998) propose a QBC strategy for classification based on *Kullback-Leibler (KL) divergence*, an information-theoretic measure of the difference between two probability distributions. The most informative query is considered to be the one with the largest average KL divergence between a committee member's posterior label distribution and the consensus. We modify this approach for sequence models by summing the average KL scores using the marginals at each token position and, as with vote entropy, normalizing for length. We call this approach **Kullback-Leibler (KL)**:

$$\phi^{KL}(\mathbf{x}) = \frac{1}{T}\sum_{t=1}^{T}\frac{1}{C}\sum_{c=1}^{C} D(\theta^{(c)} \| \mathcal{C}),$$

where (using shorthand again):

$$D(\theta^{(c)} \| \mathcal{C}) = \sum_{m=1}^{M} P_{\theta^{(c)}}(y_t = m) \log \frac{P_{\theta^{(c)}}(y_t = m)}{P_{\mathcal{C}}(y_t = m)}.$$

Here $P_{\mathcal{C}}(y_t = m) = \frac{1}{C}\sum_{c=1}^{C} P_{\theta^{(c)}}(y_t = m)$, or the "consensus" marginal probability that m is the label at position t in the sequence.

Both of these disagreement measures are normalized for sequence length T. As with token entropy (2), this may bias the learner toward querying shorter sequences. To study the effects of normalization, we also conduct experiments with **non-normalized variants** ϕ^{TVE} and ϕ^{TKL}.

Additionally, we argue that these token-level disagreement measures may be less appropriate for most tasks than measuring the committee's disagreement about the label sequence $\mathbf{y}$ as a whole. Therefore, we propose **sequence vote entropy (SVE)**:

$$\phi^{SVE}(\mathbf{x}) = -\sum_{\hat{\mathbf{y}} \in \mathcal{N}^{\mathcal{C}}} P(\hat{\mathbf{y}}|\mathbf{x};\mathcal{C}) \log P(\hat{\mathbf{y}}|\mathbf{x};\mathcal{C}),$$

where $\mathcal{N}^{\mathcal{C}}$ is the union of the N-best parses from all models in the committee $\mathcal{C}$, and $P(\hat{\mathbf{y}}|\mathbf{x};\mathcal{C}) = \frac{1}{C}\sum_{c=1}^{C} P(\hat{\mathbf{y}}|\mathbf{x};\theta^{(c)})$, or the "consensus" posterior probability for some label sequence $\hat{\mathbf{y}}$. This can be thought of as a QBC generalization of N-best entropy, where each committee member casts a vote for the posterior label distribution. We also explore a **sequence Kullback-Leibler (SKL)** variant:

$$\phi^{SKL}(\mathbf{x}) = \frac{1}{C}\sum_{c=1}^{C}\sum_{\hat{\mathbf{y}} \in \mathcal{N}^{\mathcal{C}}} P(\hat{\mathbf{y}}|\mathbf{x};\theta^{(c)}) \log \frac{P(\hat{\mathbf{y}}|\mathbf{x};\theta^{(c)})}{P(\hat{\mathbf{y}}|\mathbf{x};\mathcal{C})}.$$

3.3 Expected Gradient Length

A third general active learning framework we consider is to query the instance that would impart the greatest change to the current model *if we knew its label*. Since we train discriminative models like CRFs using gradient-based optimization, this involves querying the instance which, if labeled and added to the training set, would create the greatest change in the gradient of the objective function (i.e., the largest gradient vector used to re-estimate parameter values).

Let $\nabla\ell(\mathcal{L};\theta)$ be the gradient of the log-likelihood ℓ with respect to the model parameters θ, as given by Sutton and McCallum (2006). Now let $\nabla\ell(\mathcal{L}^{+\langle\mathbf{x},\mathbf{y}\rangle};\theta)$ be the new gradient that would be obtained by adding the training tuple $\langle\mathbf{x},\mathbf{y}\rangle$ to $\mathcal{L}$. Since the query algorithm does not know the true label sequence $\mathbf{y}$ in advance, we instead calculate the **expected gradient length (EGL)**:

$$\phi^{EGL}(\mathbf{x}) = \sum_{\hat{\mathbf{y}}\in\mathcal{N}} P(\hat{\mathbf{y}}|\mathbf{x};\theta)\left\|\nabla\ell(\mathcal{L}^{+\langle\mathbf{x},\hat{\mathbf{y}}\rangle};\theta)\right\|,$$

approximated as an expectation over the N-best labelings, where $\|\cdot\|$ is the Euclidean norm of each resulting gradient vector. We first introduced this approach in previous work on multiple-instance active learning (Settles et al., 2008), and adapt it to query selection with sequences here. Note that, at query time, $\nabla\ell(\mathcal{L};\theta)$ should be nearly zero since ℓ converged at the previous round of training. Thus, we can approximate $\nabla\ell(\mathcal{L}^{+\langle\mathbf{x},\hat{\mathbf{y}}\rangle};\theta) \approx \nabla\ell(\langle\mathbf{x},\hat{\mathbf{y}}\rangle;\theta)$ for computational efficiency, because the training instances are assumed to be independent.

3.4 Information Density

It has been suggested that uncertainty sampling and QBC are prone to querying outliers (Roy and McCallum, 2001; Zhu et al., 2003). Figure 2 illustrates this problem for a binary linear classifier using uncertainty sampling. The least certain instance lies on the classification boundary, but is not "representative" of other instances in the distribution, so knowing its label is unlikely to improve accuracy on the data as a whole. QBC and EGL exhibit similar behavior, by spending time querying possible outliers simply because they are controversial, or are expected to impart significant change in the model.

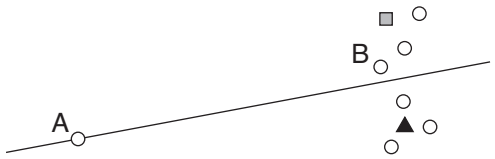

Figure 2: An illustration of when uncertainty sampling can be a poor strategy for classification. Shaded polygons represent labeled instances ($\mathcal{L}$), and circles represent unlabeled instances ($\mathcal{U}$). Since A is on the decision boundary, it will be queried as the most uncertain. However, querying B is likely to result in more information about the data as a whole.

We argue that this phenomenon can occur with sequence labeling tasks as well as with classification. To address this, we propose a new active learning approach called **information density (ID)**:

$$\phi^{ID}(\mathbf{x}) = \phi^{SE}(\mathbf{x}) \times \left(\frac{1}{U}\sum_{u=1}^{U}\text{sim}(\mathbf{x},\mathbf{x}^{(u)})\right)^{\beta}.$$

That is, the informativeness of $\mathbf{x}$ is weighted by its average similarity to all other sequences in $\mathcal{U}$, subject to a parameter β that controls the relative importance of the density term. In the formulation presented above, sequence entropy ϕ^{SE} measures the "base" informativeness, but we could just as easily use any of the instance-level strategies presented in the previous sections.

This density measure requires us to compute the similarity of two sequences. To do this, we first transform each $\mathbf{x}$, which is a sequence of feature vectors (tokens), into a single kernel vector $\vec{\mathbf{x}}$:

$$\vec{\mathbf{x}} = \left[\sum_{t=1}^{T} f_1(x_t), \ldots, \sum_{t=1}^{T} f_J(x_t)\right],$$

where $f_j(x_t)$ is the value of feature f_j for token x_t, and J is the number of features in the input representation[2]. In other words, sequence $\mathbf{x}$ is compressed into a fixed-length feature vector $\vec{\mathbf{x}}$, for which each element is the sum of the corresponding feature's values across all tokens. We can then use cosine

[2]Note that $J \neq K$, and $f_j(x_t)$ here differs slightly from the feature definition given in Section 2. Since the labels y_{t-1} and y_t are unknown before querying, the K features used for model training are reduced down to the J input features here, which factor out any label dependencies.

similarity on this simplified representation:

$$\mathrm{sim}_{cos}(\mathbf{x}, \mathbf{x}^{(u)}) = \frac{\vec{\mathbf{x}} \cdot \vec{\mathbf{x}}^{(u)}}{\|\vec{\mathbf{x}}\| \times \|\vec{\mathbf{x}}^{(u)}\|}.$$

We have also investigated similarity functions based on exponentiated Euclidean distance and KL-divergence, the latter of which was also employed by McCallum and Nigam (1998) for density-weighting QBC in text classification. However, these measures show no improvement over cosine similarity, and require setting additional hyper-parameters.

One potential drawback of information density is that the number of required similarity calculations grows quadratically with the number of instances in $\mathcal{U}$. For pool-based active learning, we often assume that the size of $\mathcal{U}$ is very large. However, these densities only need to be computed once, and are independent of the base information measure. Thus, when employing information density in a real-world interactive learning setting, the density scores can simply be pre-computed and cached for efficient lookup during the actual active learning process.

3.5 Fisher Information

We also introduce a query selection strategy for sequence models based on *Fisher information*, building on the theoretical framework of Zhang and Oles (2000). Fisher information $\mathcal{I}(\theta)$ represents the overall uncertainty about the estimated model parameters θ, as given by:

$$\mathcal{I}(\theta) = -\int_{\mathbf{x}} P(\mathbf{x}) \int_{\mathbf{y}} P(\mathbf{y}|\mathbf{x};\theta) \frac{\partial^2}{\partial \theta^2} \log P(\mathbf{y}|\mathbf{x};\theta).$$

For a model with K parameters, the Fisher information takes the form of a $K \times K$ covariance matrix. Our goal in active learning is to select the query that most efficiently minimizes the model variance reflected in $\mathcal{I}(\theta)$. This can be accomplished by optimizing the **Fisher information ratio (FIR)**:

$$\phi^{FIR}(\mathbf{x}) = -\mathrm{tr}\left(\mathcal{I}_{\mathbf{x}}(\theta)^{-1} \mathcal{I}_{\mathcal{U}}(\theta)\right), \qquad (4)$$

where $\mathcal{I}_{\mathbf{x}}(\theta)$ and $\mathcal{I}_{\mathcal{U}}(\theta)$ are Fisher information matrices for sequence $\mathbf{x}$ and the unlabeled pool $\mathcal{U}$, respectively. The leading minus sign again ensures that ϕ^{FIR} is a maximizer for use with Algorithm 1.

Previously, Fisher information for active learning has only been investigated in the context of simple binary classification. When employing FIR with sequence models like CRFs, there are two additional computational challenges. First, we must integrate over all possible labelings $\mathbf{y}$, which can, as we have seen, be approximated as an expectation over the N-best labelings. Second, the inner product in the ratio calculation (4) requires inverting a $K \times K$ matrix for each $\mathbf{x}$. In most interesting natural language applications, K is very large, making this algorithm intractable. However, it is common in similar situations to approximate the Fisher information matrix with its diagonal (Nyffenegger et al., 2006). Thus we estimate $\mathcal{I}_{\mathbf{x}}(\theta)$ using:

$$\mathcal{I}_{\mathbf{x}}(\theta) = \sum_{\hat{\mathbf{y}} \in \mathcal{N}} P(\hat{\mathbf{y}}|\mathbf{x};\theta) \left[\left(\frac{\partial \log P(\hat{\mathbf{y}}|\mathbf{x};\theta)}{\partial \theta_1} \right)^2 + \delta, \ldots, \left(\frac{\partial \log P(\hat{\mathbf{y}}|\mathbf{x};\theta)}{\partial \theta_K} \right)^2 + \delta \right],$$

and $\mathcal{I}_{\mathcal{U}}(\theta)$ using:

$$\mathcal{I}_{\mathcal{U}}(\theta) = \frac{1}{U} \sum_{u=1}^{U} \mathcal{I}_{\mathbf{x}^{(u)}}(\theta).$$

For CRFs, the partial derivative at the root of each element in the diagonal vector is given by:

$$\frac{\partial \log P(\hat{\mathbf{y}}|\mathbf{x};\theta)}{\partial \theta_k} = \sum_{t=1}^{T} f_k(\hat{y}_{t-1}, \hat{y}_t, \mathbf{x}_t) - \sum_{t=1}^{T} \sum_{y,y'} P(y, y'|\mathbf{x}) f_k(y, y', \mathbf{x}_t),$$

which is similar to the equation used to compute the training gradient, but without a regularization term. A smoothing parameter $\delta \ll 1$ is added to prevent division by zero when computing the ratio.

Notice that this method implicitly selects representative instances by favoring queries with Fisher information $\mathcal{I}_{\mathbf{x}}(\theta)$ that is not only high, but similar to that of the overall data distribution $\mathcal{I}_{\mathcal{U}}(\theta)$. This is in contrast to information density, which tries to query representative instances by explicitly modeling the distribution with a density weight.

Corpus	Entities	Features	Instances
CoNLL-03	4	78,644	19,959
NLPBA	5	128,401	18,854
BioCreative	1	175,331	10,000
FlySlip	1	31,353	1,220
CORA:Headers	15	22,077	935
CORA:References	13	4,208	500
Sig+Reply	2	25	617
SigIE	12	10,600	250

Table 1: Properties of the different evaluation corpora.

4 Empirical Evaluation

In this section we present a large-scale empirical analysis of the query strategies described in Section 3 on eight benchmark information extraction and document segmentation corpora. The data sets are summarized in Table 1.

4.1 Data and Methodology

CoNLL-03 (Sang and DeMeulder, 2003) is a collection of newswire articles annotated with four entities: person, organization, location, and misc. NLPBA (Kim et al., 2004) is a large collection of biomedical abstracts annotated with five entities of interest, such as protein, RNA, and cell-type. BioCreative (Yeh et al., 2005) and FlySlip (Vlachos, 2007) also comprise texts in the biomedical domain, annotated for gene entity mentions in articles from the human and fruit fly literature, respectively. CORA (Peng and McCallum, 2004) consists of two collections: a set of research paper headers annotated for entities such as title, author, and institution; and a collection of references annotated with BibTeX fields such as journal, year, and publisher. The Sig+Reply corpus (Carvalho and Cohen, 2004) is a set of email messages annotated for signature and quoted reply line segments. SigIE is a subset of the signature blocks from Sig+Reply which we have enhanced with several address book fields such as name, email, and phone. All corpora are formatted in the "IOB" sequence representation (Ramshaw and Marcus, 1995).

We implement all fifteen query selection strategies described in Section 3 for use with CRFs, and evaluate them on all eight data sets. We also compare against two baseline strategies: random instance selection (i.e., passive learning), and naïvely querying the longest sequence in terms of tokens. We use a typical feature set for each corpus based on the cited literature (including words, orthographic patterns, part-of-speech, lexicons, etc.). Where the N-best approximation is used $N = 15$, and for all QBC methods $C = 3$; these figures exhibited a good balance of accuracy and training speed in preliminary work. For information density, we arbitrarily set $\beta = 1$ (i.e., the information and density terms have equal weight). In each experiment, $\mathcal{L}$ is initialized with five random labeled instances, and up to 150 queries are subsequently selected from $\mathcal{U}$ in batches of size $B = 5$. All results are averaged across five folds using cross-validation.

We evaluate each query strategy by constructing learning curves that plot the overall F_1 measure (for all entities or segments) as a function of the number of instances queried. Due to lack of space, we cannot show learning curves for every experiment. Instead, Table 2 summarizes our results by reporting the area under the learning curve for all strategies on all data. Figure 3 presents a few representative learning curves for six of the corpora.

4.2 Discussion of Learning Curves

The first conclusion we can draw from these results is that there is no single clear winner. However, information density (ID), which we introduce in this paper, stands out. It usually improves upon the base sequence entropy measure, never performs poorly, and has the highest average area under the learning curve across all tasks. It seems particularly effective on large corpora, which is a typical assumption for the active learning setting. Sequence vote entropy (SVE), a QBC method we propose here, is also noteworthy in that it is fairly consistently among the top three strategies, although never the best.

Second, the top uncertainty sampling strategies are least confidence (LC) and sequence entropy (SE), the latter being the dominant entropy-based method. Among the QBC strategies, sequence vote entropy (SVE) is the clear winner. We conclude that these three methods are the best base information measures for use with information density.

Third, query strategies that evaluate the entire sequence (SE, SVE, SKL) are generally superior to those which aggregate token-level information. Furthermore, the *total* token-level strategies (TTE, TVE, TKL) outperform their *length-*

	Baselines		Uncertainty Sampling						Query-By-Committee						Other		
Corpus	**Rand**	**Long**	**LC**	**M**	**TE**	***TTE***	**SE**	**NSE**	**VE**	***KL***	***TVE***	***TKL***	***SVE***	***SKL***	***EGL***	***ID***	***FIR***
CoNLL-03	78.8	79.4	89.4	84.5	38.9	**89.7**	**90.1**	89.1	45.9	62.0	86.7	81.7	89.0	87.9	87.3	**89.6**	81.7
NLPBA	59.9	67.6	71.0	62.9	53.4	70.9	71.5	68.9	52.4	53.1	66.9	63.5	**71.8**	68.5	69.3	**73.1**	**73.6**
BioCreative	34.6	26.9	54.8	46.8	37.8	53.0	56.0	50.5	35.2	37.4	49.2	45.1	**56.6**	50.8	51.5	**59.1**	**58.8**
FlySlip	112.1	121.0	125.1	119.5	110.3	124.9	**125.4**	124.1	113.3	109.4	124.1	119.5	122.7	120.7	**125.9**	**126.8**	118.2
Headers	76.0	78.2	**81.4**	78.6	78.5	78.5	**80.8**	80.4	72.8	78.5	79.7	78.5	**80.7**	78.4	79.6	80.2	79.1
References	**90.0**	86.0	89.8	**91.5**	84.4	88.6	88.4	89.4	85.1	89.1	88.7	88.2	**89.9**	86.9	88.2	88.7	87.1
Sig+Reply	129.1	129.6	132.1	132.3	131.7	131.6	131.4	**133.1**	131.4	130.7	132.1	130.6	**132.8**	132.3	130.5	131.5	**133.2**
SigIE	84.3	82.7	88.8	87.3	89.3	88.3	87.6	89.1	**89.8**	85.5	**89.7**	85.1	**89.5**	**89.7**	87.7	88.5	88.5
Average	83.1	83.9	**91.6**	87.9	78.0	90.7	**91.4**	90.6	78.2	80.7	89.6	86.5	**91.6**	89.4	90.0	**92.2**	90.0

Table 2: Detailed results for all query strategies on all evaluation corpora. Reported is the area under the F_1 learning curve for each strategy after 150 queries (maximum possible score is 150). For each row, the **best method** is shown boxed in bold, the **second best** is shown underlined in bold, and the **third best** is shown in bold. The last row summarizes the results across all eight tasks by reporting the average area for each strategy. Query strategy formulations for sequence models introduced in this paper are indicated with italics along the top.

normalized counterparts (TE, VE, KL) in nearly all cases. In fact, the normalized variants are often inferior even to the baselines. While an argument can be made that these shorter sequences might be easier to label from a human annotator's perspective, our ongoing work indicates that the relationship between instance length and actual labeling costs (e.g., elapsed annotation time) is not a simple one. Analysis of our experiment logs also shows that length-normalized methods are occasionally biased toward short sequences with little intuitive value (e.g., sentences with few or no entities to label). In addition, vote entropy appears to be a better disagreement measure for QBC strategies than KL divergence.

Finally, Fisher information (FIR), while theoretically sound, exhibits behavior that is difficult to interpret. It is sometimes the winning strategy, but occasionally only on par with the baselines. When it does show significant gains over the other strategies, these gains appear to be only for the first several queries (e.g., NLPBA and BioCreative in Figure 3). This inconsistent performance may be a result of the approximations made for computational efficiency. Expected gradient length (EGL) also appears to exhibit mediocre performance, and is likely not worth its additional computational expense.

4.3 Discussion of Run Times

Here we discuss the execution times for each query strategy using current hardware. The uncertainty sampling methods are roughly comparable in run time (token-based methods run slightly faster), each routinely evaluating tens of thousands of sequences in under a minute. The QBC methods, on the other hand, must re-train multiple models with each query, resulting in a lag of three to four minutes per query batch (and up to 20 minutes for corpora with more entity labels).

The expected gradient length and Fisher information methods are the most computationally expensive, because they must first perform inference over the possible labelings and then calculate gradients for each candidate label sequence. As a result, they take eight to ten minutes (upwards of a half hour on the larger corpora) for each query. Unlike the other strategies, their time complexities also scale linearly with the number of model parameters K which, in turn, increases as new sequences are added to $\mathcal{L}$.

As noted in Section 3.4, information density incurs a large computational cost to estimate the density weights, but these can be pre-computed and cached for efficient lookup. In our experiments, this pre-processing step takes less than a minute for the smaller corpora, about a half hour for CoNLL-03 and BioCreative, and under two hours for NLPBA. The density lookup causes no significant change in the run time of the base information measure. Given these results, we advocate information density with an uncertainty sampling base measure in practice, particularly for active learning with large corpora.

5 Conclusion

In this paper, we have presented a detailed analysis of active learning for sequence labeling tasks. In particular, we have described and criticized the query selection strategies used with probabilistic se-

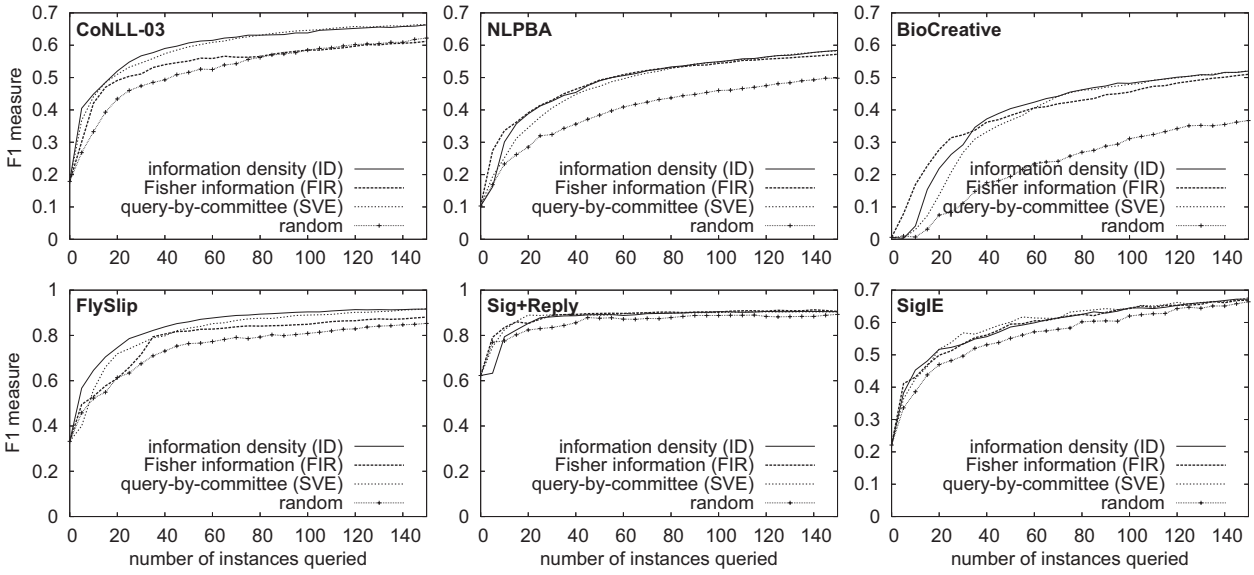

Figure 3: Learning curves for selected query strategies on six of the evaluation corpora.

quence models to date, and proposed several novel strategies to address some of their shortcomings. Our large-scale empirical evaluation demonstrates that some of these newly proposed methods advance the state of the art in active learning with sequence models. These methods include information density (which we recommend in practice), sequence vote entropy, and sometimes Fisher information.

Acknowledgments

We would like to thank the anonymous reviewers for their helpful feedback. This work was supported by NIH grants T15-LM07359 and R01-LM07050.

References

N. Abe and H. Mamitsuka. 1998. Query learning strategies using boosting and bagging. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 1–9. Morgan Kaufmann.

J. Baldridge and M. Osborne. 2004. Active learning and the total cost of annotation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9–16. ACL Press.

V.R. Carvalho and W. Cohen. 2004. Learning to extract signature and reply lines from email. In *Proceedings of the Conference on Email and Anti-Spam (CEAS)*.

D. Cohn, L. Atlas, and R. Ladner. 1994. Improving generalization with active learning. *Machine Learning*, 15(2):201–221.

A. Culotta and A. McCallum. 2005. Reducing labeling effort for stuctured prediction tasks. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, pages 746–751. AAAI Press.

I. Dagan and S. Engelson. 1995. Committee-based sampling for training probabilistic classifiers. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 150–157. Morgan Kaufmann.

R. Hwa. 2004. Sample selection for statistical parsing. *Computational Linguistics*, 30(3):73–77.

J. Kim, T. Ohta, Y. Tsuruoka, Y. Tateisi, and N. Collier. 2004. Introduction to the bio-entity recognition task at JNLPBA. In *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications (NLPBA)*, pages 70–75.

S. Kim, Y. Song, K. Kim, J.W. Cha, and G.G. Lee. 2006. MMR-based active machine learning for bio named entity recognition. In *Proceedings of Human Language Technology and the North American Association for Computational Linguistics (HLT-NAACL)*, pages 69–72. ACL Press.

J. Lafferty, A. McCallum, and F. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 282–289. Morgan Kaufmann.

K. Lari and S. J. Young. 1990. The estimation of stochastic context-free grammars using the inside-outside algorithm. *Computer Speech and Language*, 4:35–56.

D. Lewis and J. Catlett. 1994. Heterogeneous uncertainty sampling for supervised learning. In *Pro-*

ceedings of the International Conference on Machine Learning (ICML), pages 148–156. Morgan Kaufmann.

G. Mann and A. McCallum. 2007. Efficient computation of entropy gradient for semi-supervised conditional random fields. In *Proceedings of the North American Association for Computational Linguistics (NAACL)*, pages 109–112. ACL Press.

A. McCallum and K. Nigam. 1998. Employing EM in pool-based active learning for text classification. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 359–367. Morgan Kaufmann.

M. Nyffenegger, J.C. Chappelier, and E. Gaussier. 2006. Revisiting Fisher kernels for document similarities. In *Proceedings of the European Conference on Machine Learning (ECML)*, pages 727–734. Springer.

F. Peng and A. McCallum. 2004. Accurate information extraction from research papers using conditional random fields. In *Proceedings of Human Language Technology and the North American Association for Computational Linguistics (HLT-NAACL)*. ACL Press.

L. R. Rabiner. 1989. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286.

L.A. Ramshaw and M.P. Marcus. 1995. Text chunking using transformation-based learning. In *Proceedings of the ACL Workshop on Very Large Corpora*.

N. Roy and A. McCallum. 2001. Toward optimal active learning through sampling estimation of error reduction. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 441–448. Morgan Kaufmann.

E.F.T.K. Sang and F. DeMeulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Conference on Natural Language Learning (CoNLL)*, pages 142–147.

T. Scheffer, C. Decomain, and S. Wrobel. 2001. Active hidden Markov models for information extraction. In *Proceedings of the International Conference on Advances in Intelligent Data Analysis (CAIDA)*, pages 309–318. Springer-Verlag.

R. Schwartz and Y.-L. Chow. 1990. The N-best algorithm: an efficient and exact procedure for finding the N most likely sentence hypotheses. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 81–83. IEEE Press.

B. Settles, M. Craven, and S. Ray. 2008. Multiple-instance active learning. In *Advances in Neural Information Processing Systems (NIPS)*, volume 20, pages 1289–1296. MIT Press.

H.S. Seung, M. Opper, and H. Sompolinsky. 1992. Query by committee. In *Proceedings of the ACM Workshop on Computational Learning Theory*, pages 287–294.

C. E. Shannon. 1948. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423,623–656.

C. Sutton and A. McCallum. 2006. An introduction to conditional random fields for relational learning. In L. Getoor and B. Taskar, editors, *Introduction to Statistical Relational Learning*. MIT Press.

A. Vlachos. 2007. Evaluating and combining biomedical named entity recognition systems. In *BioNLP 2007: Biological, translational, and clinical language processing*, pages 199–206.

A. Yeh, A. Morgan, M. Colosimo, and L. Hirschman. 2005. Biocreative task 1a: gene mention finding evaluation. *BMC Bioinformatics*, 6(Suppl 1):S2.

T. Zhang and F.J. Oles. 2000. A probability analysis on the value of unlabeled data for classification problems. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 1191–1198. Morgan Kaufmann.

X. Zhu, J. Lafferty, and Z. Ghahramani. 2003. Combining active learning and semi-supervised learning using gaussian fields and harmonic functions. In *Proceedings of the ICML Workshop on the Continuum from Labeled to Unlabeled Data*, pages 58–65.