Scalable Active Learning by Approximated Error Reduction

Weijie Fu
Hefei University of Technology
Hefei, China
fwj.edu@gmail.com

Meng Wang
Hefei University of Technology
Hefei, China
eric.mengwang@gmail.com

Shijie Hao
Hefei University of Technology
Hefei, China
hfut.hsj@gmail.com

Xindong Wu
Hefei University of Technology
Hefei, China
xwu@hfut.edu.cn

ABSTRACT

We study the problem of active learning for multi-class classification on large-scale datasets. In this setting, the existing active learning approaches built upon uncertainty measures are ineffective for discovering unknown regions, and those based on expected error reduction are inefficient owing to their huge time costs. To overcome the above issues, this paper proposes a novel query selection criterion called approximated error reduction (AER). In AER, the error reduction of each candidate is estimated based on an expected impact over all datapoints and an approximated ratio between the error reduction and the impact over its nearby datapoints. In particular, we utilize hierarchical anchor graphs to construct the candidate set as well as the nearby datapoint sets of these candidates. The benefit of this strategy is that it enables a hierarchical expansion of candidates with the increase of labels, and allows us to further accelerate the AER estimation. We finally introduce AER into an efficient semi-supervised classifier for scalable active learning. Experiments on publicly available datasets with the sizes varying from thousands to millions demonstrate the effectiveness of our approach.

KEYWORDS

active learning, query selection, efficient algorithms

ACM Reference Format:
Weijie Fu, Meng Wang, Shijie Hao, and Xindong Wu. 2018. Scalable Active Learning by Approximated Error Reduction. In *KDD '18: The 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, August 19–23, 2018, London, United Kingdom.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3219819.3219954

KDD '18, August 19–23, 2018, London, United Kingdom

ACM ISBN 978-1-4503-5552-0/18/08…$15.00
https://doi.org/10.1145/3219819.3219954

1 INTRODUCTION

With the explosive growth of datasets, supervised learning and semi-supervised learning have been broadly used in many multi-class classification tasks, such as speech recognition [6], image classification [4], and data mining [5]. The former directly employs labeled data to train its classifier, while the latter further exploits the prior knowledge from unlabeled data to improve the classification.

To obtain satisfactory performance, classifiers require high-quality labeled data. Active learning that selects valuable queries to label has been studied to address this problem [9], [18]. The uncertainty-based sampling is the simplest query selection criterion [12]. However, as the methods based on this criterion consider each datapoint independently, they ignore the accuracy improvement on other datapoints after labeling the selected query. Although several density-weighting approaches were developed to relieve this issue [14], [25], they are still insufficiently effective to discover unknown regions, especially at the early phase of query selection.

An alternative active learning criterion called expected error reduction (EER) was therefore proposed. In general, EER makes a tradeoff on the reduction in generalization errors achieved by either labeling an unknown region or tuning decision boundaries under its current classifier, which leads to impressive performance [1], [29]. Nevertheless, EER brings a huge time cost owing to its error reduction estimation. That is, for each datapoint, the classifier has to be re-optimized with its possible labels and the labels of other datapoints need to be re-inferred to calculate its expected generalization error. As a result, even scalable classifiers are employed [16], [23], [28], the EER-based query selection is still inefficient for active learning on large-scale datasets.

To overcome the above issues, this paper proposes a novel criterion called approximated error reduction (AER). According to AER, we estimate the error reduction of a candidate based on an expected impact over all datapoints, and an approximated ratio between the error reduction and the impact over its nearby datapoints. Meanwhile, a hierarchical anchor graph [23] is utilized to build the candidate set as well as the nearby datapoint sets of these candidates. Of note, the construction of the hierarchical anchor graph is efficient, and the anchor sets and the datapoint set on this graph establish coarse-to-fine coverings of the data distribution. As

a consequence, it allows us to expand the candidate set hierarchically with the increase of labeled queries, which can further accelerate the AER estimation. Finally, by introducing the proposed AER criterion into a scalable semi-supervised classifier, we obtain an efficient and effective active learning approach for query selection on large-scale datasets. The promising results on benchmark datasets highlight the superior performance of our approach[1].

The main contributions of our work are as follows.

- We propose a novel AER criterion for query selection. Compared with EER, it enables an efficient estimation of the error reduction without re-inferring labels of massive datapoints. We also utilize a hierarchical anchor graph to construct a small candidate set, which allows us to further accelerate the AER estimation.
- We introduce the AER criterion into a scalable semi-supervised classifier for active learning. Meanwhile, we develop a fast algorithm to calculate the expected impact over all datapoints for all candidates, which can be performed via direct matrix operations rather than multiple iterations.
- We show that, apart from the similar time cost to that of the uncertainty-based sampling, the remaining time cost of our AER-based approach is independent of data sizes during the query selection. Furthermore, our AER-based approach can still achieve comparable or even higher accuracies than the EER-based approach. The experimental results on different types of datasets demonstrate the effectiveness of our approach.

The rest of this paper is organized as follows. In Section 2, we review the related work on active learning. In Section 3, we introduce the preliminaries of hierarchical anchor graphs and an efficient semi-supervised classifier. In Section 4, we propose the AER criterion and use it for scalable active learning. Section 5 validates the strengths of our approach on different-size datasets, and Section 6 concludes this paper.

2 RELATED WORK

Recent years have witnessed a number of studies on active learning for searching valuable queries and reducing manual labeling costs [15], [25]. In particular, discriminative active learning has obtained satisfactory performance in many real-world applications. Different from representative active learning that only considers the feature spaces of data distributions [2], [26], these approaches are prediction dependent and can query informative instances to facilitate the improvement of the classifier for a higher accuracy.

Uncertainty-based sampling is the simplest and most widely used discriminative criterion [13]. The methods built upon this criterion generally select the query that is the least certain, where different uncertainty measures can be used, such as entropy and ℓ_p loss. In particular, Joshi et al. [10] proposed to estimate the uncertainty by merely using the probabilities of the best and the second best classes. As these approaches are prone to outliers, some methods combining representativeness were also developed [9], [14], [21], [25]. For example, Settles et al. [21] proposed to select queries based on a density-weighting uncertainty with the cosine similarity. Li et al. [14] proposed to employ the mutual information rather than the marginal density. However, the former faces a challenge of combing two different measures, and the latter has to estimate the mutual information with a cubic time cost with respect to data sizes. Recently, Dasarathy et al. [7] proposed to select uncertain queries based on the structure of a graph, which is inefficient when the number of the datapoints along decision boundaries is large.

Instead of only considering the classifiers based on current labels, one can further exploit re-optimized classifiers by giving possible labels on unlabeled datapoints. EER therefore has become an effective query selection criterion by directly minimizing the generalization error [1], [29]. Nevertheless, it also leads to the most expensive query selection approaches, as classifiers have to be re-optimized with each possible label and the labels of massive datapoints need to be re-inferred. Although Zhu et al. [29] proposed a novel method to update the label matrix of unlabeled data, and Aodha et al. [1] presented a hierarchical subquery approach for the EER estimation, they are still impractical for large-size datasets, owing to the inevitable huge time cost of classifier initialization.

Another criterion that considers possible labels is expected model change, which selects the query with the greatest expected change on the parameters of a classifier [20]. Compared with EER, it does not require the label re-inference for datapoints, which remarkably reduces time costs. When a classifier is trained with gradient-based optimization, it is equal to select the query that creates the largest change on the gradient of the objective function [3]. However, this criterion ignores the importance of the parameters corresponding to different features, which in turn reduces its effectiveness.

In short, the above criteria either do not consider the error reduction over all datapoints, or face a huge time cost in estimating error reduction. In contrast, our AER criterion can obtain an efficient error reduction estimation, which brings significant advantages for scalable active learning.

3 PRELIMINARIES

To better present our work, we introduce the preliminaries of hierarchical anchor graphs and a scalable semi-supervised classifier. Some important notations are listed in Table 1.

3.1 Hierarchical Anchor Graph

We first introduce hierarchical anchor graphs [23]. An illustrative example of graphs is shown in Fig.1

Let $\mathcal{X}_0 \in \mathbb{R}^{N_0 \times d}$ indicate the set of datapoints, and each $\mathcal{X}_b \in \mathbb{R}^{N_b \times d}$ ($b=1,\ldots,h$) denote a small set of anchors (landmark datapoints) that roughly cover data distributions [16]. A hierarchical anchor graph can be constructed based on the following constraints: (1). *Fine-to-Coarse Coverings.* The sets of anchors share the same feature space of the datapoint set, and their sizes are gradually reduced with $N_1 > \ldots > N_h$.

[1] Both the datasets and codes are available at http://github.com/fuweijie/AER

Table 1: Notations and Definitions

Notation	Definition
$\mathcal{X}_0$	The set of datapoints with the dimension d.
N_0	The number of datapoints.
h	The number of anchor sets.
$\mathcal{X}_b$	The b-th set of anchors ($h \geq b \geq 1$).
N_b	The number of anchors in $\mathcal{X}_b$ ($h \geq b \geq 1$).
C	The number of classes.
$\mathbf{Z}^{b-1,b}$	The inter-set adjacency matrix between $\mathcal{X}_{b-1}$ and $\mathcal{X}_b$.
$\mathbf{Z}^{\mathrm{H}}$	The cascaded inter-set adjacency matrix.
$\lceil \cdot \rceil$	The nearest points in the connected coarser set.
$\mathbf{A}$	The soft label matrix of the coarsest anchor set.
$\mathbf{A}^{+\tilde{y}_{qr}}$	The updated matrix with an extra label r on $\mathbf{x}_q$.
$\mathbf{F}$	The soft label matrix of the datapoint set.
$\mathbf{Y}_{\mathrm{L}}$	The class indicator matrix on labeled datapoints.
$\bar{\mathcal{E}}$	The average estimated error based on labeled data.
$\mathcal{S}_{\mathrm{AL}}$	The set of candidates for active learning.
N_q	The number of candidates in $\mathcal{S}_{\mathrm{AL}}$.
$\mathcal{I}_q$	The expected impact over all datapoints of $\mathbf{x}_q$.
$\langle q \rangle$	The indices of the nearby datapoints of $\mathbf{x}_q$.
$N_{\langle q \rangle}$	The number of the nearby datapoints of $\mathbf{x}_q$.
$\mathcal{E}r_{\langle q \rangle}$	The error reduction over the nearby datapoints of $\mathbf{x}_q$.
$\mathbb{E}(\mathcal{E}^{+\tilde{y}_{qr}}_{\langle q \rangle})$	The generalization error over the nearby datapoints.
$\mathcal{I}_{\langle q \rangle}$	The impact over the nearby datapoints of $\mathbf{x}_q$.
$\lfloor q \rfloor$	The set of points whose nearest anchor in the connected coarser set is $\mathbf{x}_q$.

These anchor sets bring fine-to-coarse coverings of the data distribution. (2). *Pyramidal Structure.* Let $\mathcal{G}$ denote a multiple-set pyramidal graph. The original datapoints in $\mathcal{X}_0$ locate at the bottom layer of the pyramid, and the remaining layers are all composed of fine-to-coarse anchor sets with $\mathcal{X}_1,\ldots,\mathcal{X}_h$. (3). *Inter-set Adjacency.* The datapoint set and all anchor sets are linked up to a complete graph with h sets of inter-set adjacency edges between the neighboring sets, such as $\mathbf{Z}^{b-1,b} \in \mathbb{R}^{N_{b-1} \times N_b}$ between $\mathcal{X}_{b-1}$ and $\mathcal{X}_b$.

In the above graph model, we denote anchors in $\mathcal{X}_1$ and $\mathcal{X}_h$ as the finest anchors and the coarsest anchors, respectively. In particular, if the graph only contains one anchor set (h=1), we denote anchors in $\mathcal{X}_1$ as the coarsest anchors for convenience. Besides, we use 'N_1-N_2-...-N_h-anchor-graph' to indicate a hierarchical anchor graph built upon h anchor sets with $N_1,N_2,\ldots,N_h$ anchors in different anchor sets. More details of the setting of anchor sets can be found in [23].

The remaining issues of the graph construction involve two aspects, including the generation of anchor sets and the inter-set adjacency estimation between the neighboring sets. To obtain an anchor set, we can perform a fast clustering algorithm on the datapoint set with a predetermined number of centers [16]. Here we briefly describe the adjacency estimation. Specifically, for each $\mathbf{Z}^{b-1,b} \in \mathbb{R}^{N_{b-1} \times N_b}$, its weights can be determined by the kernel regression [16]:

$$Z_{ij}^{b-1,b} = \frac{K_\sigma(\mathbf{x}_i, \mathbf{x}_j)}{\sum_{j' \in \lceil i \rceil} K_\sigma(\mathbf{x}_i, \mathbf{x}_{j'})}, \ \forall j \in \lceil i \rceil, \qquad (1)$$

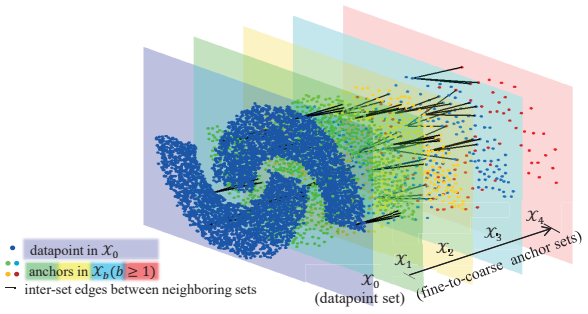


Figure 1: An example of hierarchical anchor graphs.

where σ is the bandwidth of the Gaussian kernel, $\mathbf{x}_i$ is the i-th point in $\mathcal{X}_{b-1}$, and $\lceil i \rceil$ is the set of indices of its s nearest anchors in the connected coarser set $\mathcal{X}_b$. For large-scale datasets, we can speed up the weight estimation with the approximate nearest neighbor search [17], which reduces the cost of the graph construction to $O(N_0 \log N_1)$.

3.2 Scalable Semi-Supervised Learning

Efficient semi-supervised classifiers have been proposed for large-scale classification [16], [23], [28]. Here we introduce a scalable one built upon the above graph, which has shown its effectiveness on many multi-class classification tasks [23].

Let $\mathcal{D}=\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_{N_{\mathrm{L}}}, y_{N_{\mathrm{L}}}), \ldots, \mathbf{x}_{N_0}\}$ be the dataset where the first N_{L} datapoints are labeled from C distinct classes. Let $\mathbf{A}$ denote the soft label matrix on the anchors in $\mathcal{X}_h$ that needs to be optimized. Based on the inter-set adjacency in the hierarchical anchor graph, the soft label matrix on datapoints ($\mathbf{F}$) can be inferred in a hierarchical manner: $\mathbf{F}=\mathbf{Z}^{\mathrm{H}}\mathbf{A} \in \mathbb{R}^{N_0 \times C}$, where $\mathbf{Z}^{\mathrm{H}}=\mathbf{Z}^{0,1}(\ldots(\mathbf{Z}^{h-1,h})) \in \mathbb{R}^{N_0 \times N_h}$ is the cascaded inter-set matrix between $\mathcal{X}_0$ and $\mathcal{X}_h$. Let $\mathbf{\Lambda}$ be a diagonal matrix with $\Lambda_{jj}=\sum_{i=1}^{N_0} Z_{ij}^{0,1}$, and $\mathbf{rL}=\mathbf{Z}^{\mathrm{H}^{\mathrm{T}}}\mathbf{Z}^{\mathrm{H}} - \mathbf{Z}^{\mathrm{H}^{\mathrm{T}}}\mathbf{Z}^{0,1}\mathbf{\Lambda}^{-1}\mathbf{Z}^{0,1^{\mathrm{T}}}\mathbf{Z}^{\mathrm{H}}$ be the reduced Laplacian matrix on the graph. Besides, denote $\mathbf{Y}_{\mathrm{L}}=[\mathbf{y}_1;\ldots;\mathbf{y}_{N_{\mathrm{L}}}] \in \mathbb{R}^{N_{\mathrm{L}} \times C}$ as the class indicator matrix of the labeled data, where y_{ir}=1 if $\mathbf{x}_i$ belongs to the class r, and y_{ir}=0 otherwise. Let $\mathbf{Z}_{\mathrm{L}}^{\mathrm{H}}$ be the labeled part of $\mathbf{Z}^{\mathrm{H}}$, and λ be the parameter that weighs the fitting constraint against the smoothness constraint in manifold regularization. Hierarchial anchor graph regularization (HAGR) obtains an optimal solution of $\mathbf{A}$ in a closed form:

$$\mathbf{A} = (\mathbf{Z}_{\mathrm{L}}^{\mathrm{H}^{\mathrm{T}}}\mathbf{Z}_{\mathrm{L}}^{\mathrm{H}} + \lambda \mathbf{rL})^{-1}\mathbf{Z}_{\mathrm{L}}^{\mathrm{H}^{\mathrm{T}}}\mathbf{Y}_{\mathrm{L}} \in \mathbb{R}^{N_h \times C} \qquad (2)$$

with a time cost of $O({N_h}^3)$, where N_h is the size of $\mathcal{X}_h$.

Finally, HAGR employs the soft labels of the anchors in $\mathcal{X}_h$ to infer the label of any unlabeled datapoint in $\mathcal{X}_0$:

$$\mathrm{argmax}_{r \in \{1,\ldots,C\}} \frac{\mathbf{Z}_{i\cdot}^{\mathrm{H}} \times \mathbf{A}_{\cdot r}}{\pi_r}, i = N_{\mathrm{L}}+1, \ldots, N_0, \qquad (3)$$

where $\mathbf{A}_{\cdot r}$ is the r-th column of $\mathbf{A}$, and $\pi_r=\mathbf{1}^{\mathrm{T}}\mathbf{Z}^{\mathrm{H}}\mathbf{A}_{\cdot r}$ is the normalization factor [16]. As we have obtained $\mathbf{Z}^{\mathrm{H}}$, this label inference can be performed with a time cost of $O(N_0 N_h C)$.

4 PROPOSED APPROACH

In Section 4.1, we propose a novel query selection criterion called approximated error reduction (AER). We also introduce its implementing details based on hierarchical anchor graphs for scalable active learning. In Section 4.2, we present our AER-based approach. Section 4.3 analyzes its time cost, and Section 4.4 makes a comparison to other criteria.

4.1 Approximated Error Reduction

In AER, to obtain valuable queries, the error reduction of a candidate is estimated based on an expected impact over all datapoints and an approximated ratio between the error reduction and the impact over its nearby datapoints.

Suppose $\mathbf{f}_i$ is the soft label assignment of $\mathbf{x}_i$ based on current labeled datapoints, and $\widehat{\mathbf{f}}_i$ is the hard indicator vector with $\widehat{f}_{ir}$=1 if r=argmax$_r f_{ir}$ and $\widehat{f}_{ir}$=0 otherwise. Let $\mathcal{S}_{\text{AL}}$ be the candidate set. For each candidate $\mathbf{x}_q \in \mathcal{S}_{\text{AL}}$, we first calculate its expected impact over all datapoints:

$$\mathcal{I}_q = \sum_{r=1}^{C} f_{qr} \sum_{i=1}^{N_0} \ell(\mathbf{f}_i, \mathbf{f}_i^{+\hat{y}_{qr}}), \tag{4}$$

where $\mathbf{f}_i^{+\hat{y}_{qr}}$s are the re-inferred soft labels based on the current labeled datapoints and $\mathbf{x}_q$ with the label r, and ℓ denotes the l_2 loss. As we can see, $\mathcal{I}_q$ calculates the change on the soft labels of all datapoints by assuming new labels on $\mathbf{x}_q$.

Then, we consider the ratio between the error reduction and the impact of $\mathbf{x}_q$. Instead of an exact ratio built upon all datapoints, AER only requires an approximated one for $\mathbf{x}_q$ based on its error reduction and impact over nearby datapoints. Let $\langle q \rangle$ be the indices of the nearby datapoints with the size of $N_{\langle q \rangle}$. The approximated ratio can be calculated:

$$\frac{\mathcal{E}r_{\langle q \rangle}}{\mathcal{I}_{\langle q \rangle}} = \frac{\mathcal{E}_{\langle q \rangle} - \mathbb{E}(\mathcal{E}_{\langle q \rangle}^{+\hat{y}_{qr}})}{\mathcal{I}_{\langle q \rangle}}, \tag{5}$$

where

$$\begin{cases} \mathcal{E}_{\langle q \rangle} = \sum_{i=1}^{N_{\langle q \rangle}} \ell(\mathbf{f}_i, \widehat{\mathbf{f}}_i), \\ \mathcal{I}_{\langle q \rangle} = \sum_{r=1}^{C} f_{qr} \sum_{i=1}^{N_{\langle q \rangle}} \ell(\mathbf{f}_i, \mathbf{f}_i^{+\hat{y}_{qr}}), \\ \mathbb{E}(\mathcal{E}_{\langle q \rangle}^{+\hat{y}_{qr}}) = \sum_{r=1}^{C} f_{qr} \sum_{i=1}^{N_{\langle q \rangle}} \ell(\mathbf{f}_i^{+\hat{y}_{qr}}, \widehat{\mathbf{f}}_i^{+\hat{y}_{qr}}), \end{cases}$$

are the accumulated estimated error, the impact and the generalization error over these nearby datapoints, respectively.

As $\frac{\mathcal{E}r_{\langle q \rangle}}{\mathcal{I}_{\langle q \rangle}}$ is estimated based on nearby datapoints, its confidence is lower than that built upon all datapoints. Therefore, instead of applying the same confidence to $\mathcal{I}_q$ and $\frac{\mathcal{E}r_{\langle q \rangle}}{\mathcal{I}_{\langle q \rangle}}$, the objective function of the AER criterion is formulated as

$$\text{argmax}_{\mathbf{x}_q} \mathcal{I}_q \times \left(\frac{\mathcal{E}r_{\langle q \rangle}}{\mathcal{I}_{\langle q \rangle}}\right)^{1-\epsilon}, \ \mathbf{x}_q \in \mathcal{S}_{\text{AL}}, \tag{6}$$

where $\epsilon \in (0{:}1)$ aims to control the confidence of $\frac{\mathcal{E}r_{\langle q \rangle}}{\mathcal{I}_{\langle q \rangle}}$. Of note, the above formulation leads to the following conclusion.

Proposition.1: Suppose $\frac{\mathcal{E}r_{\langle q \rangle}}{\mathcal{I}_{\langle q \rangle}} > 0$ and $\mathcal{I}_q > 0$. For Eq.6 with $\epsilon \in (0, 1)$, the influence of $\frac{\mathcal{E}r_{\langle q \rangle}}{\mathcal{I}_{\langle q \rangle}}$ on the objective value is reduced, and that of $\mathcal{I}_q$ is relatively increased.

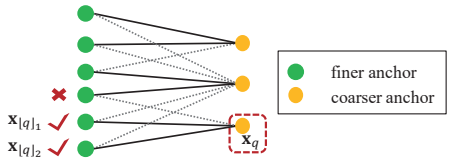


Figure 2: An example of $\mathcal{S}_{\lfloor q \rfloor}$ that denotes the set of finer anchors whose nearest connected coarser anchor is $\mathbf{x}_q$. For simplification, only a tiny fraction of inter-set edges of the graph are shown.

We leave the proof of the proposition to the Appendix. Besides, when ϵ is closer to 1, the reduction will be larger. For example, if ϵ=1, the influence of $\frac{\mathcal{E}r_{\langle q \rangle}}{\mathcal{I}_{\langle q \rangle}}$ will be zero.

In this paper, we simply set ϵ to the average estimated error as $\epsilon = \bar{\mathcal{E}} = \frac{1}{N_0} \sum_{i=1}^{N_0} \ell(\mathbf{f}_i, \widehat{\mathbf{f}}_i)$. The idea behind is that, when the error is large, the classification result is often instable. As a result, labeled candidates can affect more datapoints rather than their nearby ones, which reduces the confidence of the approximated ratio. Later we show its effectiveness via the experimental comparison to another strategy [14].

In short, for each $\mathbf{x}_q$ in $\mathcal{S}_{AL}$, if its expected impact over all datapoints ($\mathcal{I}_q$) and its ratio over nearby datapoints ($\frac{\mathcal{E}r_{\langle q \rangle}}{\mathcal{I}_{\langle q \rangle}}$) can be efficiently obtained, we can perform scalable active learning via Eq.6. Below we introduce these implementing details based on hierarchical anchor graphs.

4.1.1 *Hierarchical Expansion of Candidates ($\mathcal{S}_{AL}$)*

We first introduce a hierarchical expansion technique to construct the candidate set by employing both the coarse-to-fine anchors and datapoints on a hierarchical anchor graph.

Instead of using all unlabeled datapoints as candidates, we initialize a candidate set with the coarsest anchors in $\mathcal{X}_h$:

$$\mathcal{S}_{\text{AL}} \Leftarrow \mathcal{X}_h. \tag{7}$$

As the size of $\mathcal{X}_h$ is much smaller than that of the unlabeled data, it can significantly reduce the time cost of the AER estimation. Besides, this candidate set is sufficiently representative for query selection at the early stage.

With the increase of labels, a finer candidate set is needed to tune decision boundaries. Let $\mathcal{S}_{\lfloor q \rfloor} = \{\mathbf{x}_{\lfloor q \rfloor_1}, \ldots\}$ be the set of finer points whose nearest connected coarser anchor is $\mathbf{x}_q$. After $\mathbf{x}_q \in \mathcal{S}_{\text{AL}}$ is labeled, we expand candidates with $\mathcal{S}_{\lfloor q \rfloor}$:

$$\mathcal{S}_{\text{AL}} \Leftarrow (\mathcal{S}_{\text{AL}} \ominus \mathbf{x}_q) \cup \mathcal{S}_{\lfloor q \rfloor}, \tag{8}$$

where $\mathcal{S}_{\text{AL}} \ominus \mathbf{x}_q$ denotes the operation that removes $\mathbf{x}_q$ from $\mathcal{S}_{\text{AL}}$. Different from employing all connected finer points, we alleviate the rapid expansion of the candidate set based on a few candidates in $\mathcal{S}_{\lfloor q \rfloor}$. An illustrative example is shown in Fig.2, where only $\mathbf{x}_{\lfloor q \rfloor_1}$, $\mathbf{x}_{\lfloor q \rfloor_2}$ are added into the candidate set, while the other connected finer points of $\mathbf{x}_q$ are ignored.

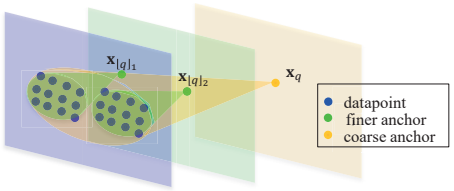


Figure 3: An example of the hierarchical assignment of nearby datapoints. In this example, the query $\mathbf{x}_q$ is labeled and $\mathbf{x}_{\lfloor q \rfloor_1}$, $\mathbf{x}_{\lfloor q \rfloor_2}$ are added into the candidate set. We only assign the datapoints in $\langle q \rangle$ to their approximate nearest candidate $\mathbf{x}_{\lfloor q \rfloor_1}$ or $\mathbf{x}_{\lfloor q \rfloor_2}$.

4.1.2 Hierarchical Assignment of Nearby Datapoints ($\langle q \rangle$)

Then, we consider the assignment of the nearby datapoints. Similar to the candidate expansion, we build the nearby datapoint sets for all candidates in a hierarchical manner.

Specifically, we first build nearby sets for the candidates in $\mathcal{X}_h$ by assigning all datapoints in $\mathcal{X}_0$ to their approximate nearest candidates. Denoting $\langle q \rangle$ as the nearby datapoints of the q-th candidate, we have:

$$\begin{cases} \langle 1 \rangle \cup \langle 2 \rangle \cup \ldots = \mathcal{X}_0, \\ \langle 1 \rangle \cap \langle 2 \rangle \cap \ldots = \oslash. \end{cases}$$

When a candidate $\mathbf{x}_q \in \mathcal{S}_{\mathrm{AL}}$ is labeled and $\mathcal{S}_{\lfloor q \rfloor}$ is added into the candidate set, we re-build nearby sets for these new candidates in $\mathcal{S}_{\lfloor q \rfloor}$ in a similar way based on the nearby datapoints of $\mathbf{x}_q$ (See Fig.3). As a consequence, we obtain:

$$\begin{cases} \langle \lfloor q \rfloor_1 \rangle \cup \langle \lfloor q \rfloor_2 \rangle \cup \ldots = \langle q \rangle, \\ \langle \lfloor q \rfloor_1 \rangle \cap \langle \lfloor q \rfloor_2 \rangle \cap \ldots = \oslash. \end{cases}$$

In general, for each finer candidate, the number of its nearby datapoints is smaller than those of coarser ones.

4.1.3 Fast Computation of the Expected Impact($\mathcal{I}_q$)

Here, we focus on the computation of the expected impact over all datapoints. Let $\mathbf{A}^{+\hat{y}_{qr}}$ denote the updated classifier with an extra label r on $\mathbf{x}_q$. Taking HAGR as an example with $\mathbf{F}=\mathbf{Z}^{\mathrm{H}}\mathbf{A}$, we can therefore obtain:

$$\mathcal{I}_q = \sum_{r=1}^{C} f_{qr} \| \mathbf{Z}^{\mathrm{H}}(\mathbf{A}^{+\hat{y}_{qr}} - \mathbf{A}) \|_{\mathrm{F}}^2. \tag{9}$$

To calculate Eq.9, we focus on its Frobenius norm:

$$\mathrm{trace}((\mathbf{A}^{+\hat{y}_{qr}} - \mathbf{A})^{\mathrm{T}} \boldsymbol{\Delta} (\mathbf{A}^{+\hat{y}_{qr}} - \mathbf{A})), \tag{10}$$

where $\boldsymbol{\Delta}=\mathbf{Z}^{\mathrm{H}\mathrm{T}}\mathbf{Z}^{\mathrm{H}}$. As $\boldsymbol{\Delta}$ only needs to be calculated once, the time cost here is much smaller than that of the generalization error in EER, where the labels of all datapoints must be incrementally re-inferred. However, as the time cost of each $\mathbf{A}^{+\hat{y}_{qr}}$ scales as $O(N_h^3)$, for N_q candidates within C classes, the total cost of $O(N_h^3 N_q C)$ can still be expensive.

The remaining challenge is how to compute Eq.10 for all candidates efficiently. To solve this issue, we present an efficient method for the impact estimation in the Appendix, which only involves fast matrix operations. In this way, the time cost can be drastically reduced to $O(N_h^2 N_q)$.

4.1.4 Fast Estimation of the Approximated Ratio ($\frac{\mathcal{E}r_{\langle q \rangle}}{\mathcal{I}_{\langle q \rangle}}$)

Now, we estimate the ratio between the error reduction and the impact over nearby datapoints. A naive solution is to calculate the two involved terms directly. However, it is practically inefficient to incrementally calculate them for all candidates or store all the relevant matrices in the memory, such as $\boldsymbol{\Delta}_{\langle q \rangle}=\mathbf{Z}^{\mathrm{H}}_{\langle q \rangle}{}^{\mathrm{T}}\mathbf{Z}^{\mathrm{H}}_{\langle q \rangle}$ for the impact estimation over nearby datapoints, where $\mathbf{Z}^{\mathrm{H}}_{\langle q \rangle}$ denotes the nearby part of $\mathbf{x}_q$ in $\mathbf{Z}^{\mathrm{H}}$.

Below we introduce an alternative to accelerate this step. First, for the candidate $\mathbf{x}_q$, we approximate its expected impact over its nearby datapoints ($\mathcal{I}_{\langle q \rangle}$) based on its expected impact over all datapoints ($\mathcal{I}_q$):

$$\mathcal{I}_{\langle q \rangle} \approx \frac{\mathcal{I}_q}{1+\mu}, \tag{11}$$

in which the auxiliary parameter $\mu \geq 0$ describes the degree that the impact overflows the nearby datapoints.

Then we consider the error reduction over nearby datapoints ($\mathcal{E}r_{\langle q \rangle}$). Instead of the repeated calculation, we evaluate it based on the estimated errors of nearby datapoints:

$$\mathcal{E}r_{\langle q \rangle} = \sum_{i=1}^{N_{\langle q \rangle}} \eta_i \times \ell(\mathbf{f}_i, \widehat{\mathbf{f}}_i), \quad i \in \langle q \rangle, \tag{12}$$

where the auxiliary parameter $\eta_i \in [0{:}1]$. It describes the degree that the expected error of the i-th nearby datapoint will be reduced from its current estimated error. Since these parameters of the nearby datapoints of each candidate η_is tend to be similar, we can approximate Eq.12 with the accumulated error over these nearby datapoints ($\mathcal{E}_{\langle q \rangle}$) in Eq.5:

$$\mathcal{E}r_{\langle q \rangle} \approx \eta \times \sum_{i=1}^{N_{\langle q \rangle}} \ell(\mathbf{f}_i, \widehat{\mathbf{f}}_i) = \eta \times \mathcal{E}_{\langle q \rangle}. \tag{13}$$

Note that the auxiliary parameters μ and η can be different for each candidate, which may involve complex relationships with respect to the uncertainty over their nearby datapoints. However, such an evaluation is difficult to be described. To circumvent this problem, in this work, we propose to apply the same settings for all candidates. Our reasoning here is as follows: as the candidate set is expanded hierarchically, we expect that the influence on the nearby datapoints of one candidate will not be much larger than that of the other candidates. With this simplification, based on Eq.11 and Eq.13, we can directly obtain the ratio in Eq.6:

$$\frac{\mathcal{E}r_{\langle q \rangle}}{\mathcal{I}_{\langle q \rangle}} = (\eta \times \mathcal{E}_{\langle q \rangle}) / (\frac{\mathcal{I}_q}{1+\mu}) = \eta(1+\mu) \times \frac{\mathcal{E}_{\langle q \rangle}}{\mathcal{I}_q}. \tag{14}$$

4.2 Scalable Active Learning

Now we present the AER-based scalable active learning.

Given a dataset $\mathcal{D}$ with N_{L} labeled datapoints, we first construct a hierarchical anchor graph with Eq.1 and build a candidate set $\mathcal{S}_{\mathrm{AL}}$ via Eq.7. We initialize the classifier based on all datapoints with a few labels via Eq.2, and infer the labels of unlabeled datapoints with Eq.3.

Then for each candidate $\mathbf{x}_q \in \mathcal{S}_{\mathrm{AL}}$, we compute its expected impact over all datapoints ($\mathcal{I}_q$) based on the proposed fast

Table 2: Our Approach for Scalable Active Learning

Input: datapoint set $\mathcal{X}_0$, anchor sets $\mathcal{X}_b$s, the parameters s and λ, the number of labeled datapoints T.
Initialization
1. Construct a hierarchical anchor graph with Eq.1.
2. Build a set of candidates $\mathcal{S}_{\mathrm{AL}}$ with Eq.7 and find the nearby datapoint sets of these candidates.
3. Initialize a scalable classifier, such as HAGR.
Efficient Query Selection
Repeat the following steps until T queries are labeled:
1. Obtain the estimated value in Eq.16 for each candidate based on the proposed fast algorithm and the labels of datapoints.
2. Ask an oracle for the label of the selected query.
3. Re-train the classifier via Eq.2 and re-infer the labels of datapoints via Eq.3.
4. Expand the candidate set $\mathcal{S}_{\mathrm{AL}}$ via Eq.8.
Output: The classifier and the labels of all datapoints.

algorithm mentioned in Section 4.1.3. We employ its nearby datapoints to estimate its approximated ratio between the error reduction and the impact ($\frac{\mathcal{E}r_{\langle q\rangle}}{\mathcal{I}_{\langle q\rangle}}$). We substitute these two terms of $\mathbf{x}_q$ into Eq.6 and obtain its approximated error reduction over all datapoints ($\mathcal{E}r_q$):

$$\begin{aligned}\mathcal{E}r_q &= \mathcal{I}_q \times \left(\frac{\mathcal{E}r_{\langle q\rangle}}{\mathcal{I}_{\langle q\rangle}}\right)^{1-\epsilon} \\ &= \mathcal{I}_q \times \left(\eta(1+\mu) \times \frac{\mathcal{E}_{\langle q\rangle}}{\mathcal{I}_q}\right)^{1-\epsilon} \\ &= \mathcal{I}_q^{\epsilon} \times {\mathcal{E}_{\langle q\rangle}}^{1-\epsilon} \times (\eta(1+\mu))^{1-\epsilon}, \end{aligned} \tag{15}$$

where ϵ=$\bar{\mathcal{E}}$. As $(\eta(1+\mu))^{1-\epsilon}$ in Eq.15 is a constant for all candidates, it can be removed without changing the solution of the optimization problem. Therefore, according to AER, the following query can be selected:

$$\mathrm{argmax}_{\mathbf{x}_q}\ \mathcal{I}_q^{\epsilon} \times {\mathcal{E}_{\langle q\rangle}}^{1-\epsilon}, \mathbf{x}_q \in \mathcal{S}_{\mathrm{AL}}. \tag{16}$$

Once the query is labeled, we re-infer the labels of datapoints and update candidates via Eq.8. In our experiments, we conduct the query selection until T queries are labeled.

The overall active learning approach is given in Table 2.

4.3 Computational Cost Analysis

Below we analyze the time cost of the proposed approach.

During the initialization step, the time cost of graph construction is $O(N_0 \log N_1)$. The total cost of computing $\mathbf{Z}^{\mathrm{H}}$, $\mathbf{\Delta}$, and $\mathbf{rL}$ scales as $O(N_0 N_h s)$ with the sparse matrix multiplication [27]. We optimize HAGR with a cost of $O(N_h^3)$. In short, the time cost here scales as $O(N_0 \log N_1 + N_0 N_h s + N_h^3)$.

Then, in each iteration of query selection, the following time costs are required. Firstly, we infer the labels of datapoints in $O(N_0 N_h C)$, and calculate their estimated errors in $O(N_0 C)$. Secondly, to estimate the error reduction for N_q candidates, we compute their expected impact values over all datapoints in $O(N_h^2 N_q)$ and the approximated ratios in $O(N_0 + N_q)$. Finally, we select the query based on AER in $O(N_q)$. As we have $N_q \geq N_h$, the time cost here can be simplified as $O(N_0 N_h C + N_h^2 N_q + N_h^3) \approx O(N_0 N_h C + N_h^2 N_q)$.

As we can see, apart from the similar time cost to that of the uncertainty-based sampling, namely $O(N_0 N_h C)$, the remaining time cost of our AER-based query selection is independent of data sizes, which highlights its superiority for large-scale active learning.

4.4 Discussion on AER

In this section, we discuss the relationships and differences between the proposed AER criterion and other criteria.

4.4.1 Comparison to Density-Weighting Uncertainty

This learning criterion [21] selects the datapoint that is both uncertain and representative, which can be formulated as

$$\mathrm{argmax}_{\mathbf{x}_q} d(\mathbf{x}_q) \times \ell^{NSE}(\mathbf{f}_q), \tag{17}$$

where $d(\mathbf{x}_q)$=$\sum_{i=1}^{N_{\mathrm{U}}} \mathrm{sim}_{cos}(\mathbf{x}_q, \mathbf{x}_i)$ denotes the cosine similarity of $\mathbf{x}_q$ over N_{U} unlabeled datapoints, and $\ell^{NSE}(\mathbf{f}_q)$ denotes the N-best sequence entropy [10]. Eq.17 can be rewritten as

$$\mathrm{argmax}_{\mathbf{x}_q} d(\mathbf{x}_q) \times \frac{\ell^{NSE}(\mathbf{f}_q)}{\mathrm{sim}_{cos}(\mathbf{x}_q, \mathbf{x}_q)}, \tag{18}$$

where $\mathrm{sim}_{cos}(\mathbf{x}_q, \mathbf{x}_q)$=1. Similar to Eq.6, the first term evaluates the similarity of $\mathbf{x}_q$ over massive datapoints, and the second term is the approximated ratio between the uncertainty reduction and the similarity of $\mathbf{x}_q$ itself. Compared with AER, Eq.18 estimates the ratio based on a single datapoint, which can be insufficiently effective.

4.4.2 Comparison to Expected Model Change

This learning criterion [3] selects the datapoint that brings the greatest expected change on the parameters of a classifier. For HAGR, its query selection can be formulated as

$$\mathrm{argmax}_{\mathbf{x}_q} \sum_{r=1}^{C} f_{qr} \| \mathbf{A}^{+\hat{y}_{qr}} - \mathbf{A} \|_{\mathrm{F}}^2. \tag{19}$$

The change of HAGR is equal to the change on the soft labels of the coarsest anchors, which is far from the error reduction over massive datapoints. In contrast, our impact in Eq.9 calculates the change on the soft labels of all datapoints, which narrows the above gap by introducing data distributions.

4.4.3 Comparison to Expected Error Reduction

In EER, if labeling the candidate $\mathbf{x}_q$ only reduces the errors of its nearby datapoints $\langle q\rangle$, we have $\mathcal{E}r_q$=$\mathcal{E}r_{\langle q\rangle}$. Below we show that when μ=0 and η=1, our AER value is equal to EER, which is independent of the average estimated error $\bar{\mathcal{E}}$.

Since η=1, we obtain $\mathcal{E}_{\langle q\rangle}$=$\mathcal{E}r_{\langle q\rangle}$ via Eq.13. As all errors of nearby datapoints are changed to 0, we obtain $\mathcal{I}_{\langle q\rangle}$=$\mathcal{E}r_{\langle q\rangle}$. Then as μ=0, we have $\mathcal{I}_q$=$\mathcal{I}_{\langle q\rangle}$ via Eq.11. By substituting the above results into Eq.15, we finally obtain:

$$\mathcal{E}r_q = \mathcal{I}_q^{\epsilon} \times {\mathcal{E}_{\langle q\rangle}}^{1-\epsilon} \times 1 = {\mathcal{E}r_{\langle q\rangle}}^{\epsilon} \times {\mathcal{E}r_{\langle q\rangle}}^{1-\epsilon} = \mathcal{E}r_{\langle q\rangle}. \tag{20}$$

When $\eta \neq 1$, and $\mu \neq 0$, AER first focuses on the expected impact over all datapoints for rapid accuracy improvements, and then adaptively pays attention to the error reduction over a few nearby datapoints for tuning decision boundaries.

Table 3: Details of the Datasets in Our Experiments.

Dataset	Num of instances	Num of categories	Num of dimensions
Alphadigits	1,404	36	320
Semeion	1,593	10	256
USPS	7,291	10	256
ISOLET	7,797	26	617
Letter	20,000	26	16
MNIST	70,000	10	784
ImageNet	256,091	200	512
MNIST8M	8,100,000	10	86

5 EXPERIMENTS

In this section, we investigate the effectiveness of our AER criterion. The experiments are implemented on a PC with i7-5820K CPU @ 3.30GHz and 64G RAM. We use the following datasets with varying sizes, including Alphadigits[2], Semeion[3], ISOLET[4], Letter[3], USPS[5], MNIST[6], ImageNet[7], and MNIST8M[23]. Some statistics of them are listed in Table 3. For convenience, we regard the first six as medium-size datasets, and the last two as large-size datasets.

5.1 Comparison to Other Methods

We first compare the proposed AER-based approach with the methods built upon several state-of-the-art active learning criteria. For scalability and fair comparisons, we use HAGR as the classifier for all active learners, which has shown its impressive performance on semi-supervised classification tasks. The methods for comparison are as follows:

1. Random Sampling: This method randomly selects queries for labeling. We denote it as 'Q_R'.

2. Maximal Uncertainty: This method selects queries based on the 2-best sequence entropy [10]. We denote it as 'Q_U'.

3. Maximal Density-Weighting Uncertainty: This method selects queries based on the density-weighting entropy with the cosine similarity [21]. We denote it as 'Q_{DWU}'

4. Maximal Expected Model Change: It selects queries based on the expected change on the parameters of a classifier [3]. This method is denoted as 'Q_{EMC}'.

5. Maximal Expected Impact: This method selects queries based on the proposed expected impact over all datapoints. We denote it as 'Q_{EI}'.

6. Maximal EER: This method selects queries with the EER evaluation [29]. We denote it as 'Q_{EER}'.

7. Maximal AER: This proposed method chooses queries based AER. It is denoted as 'Q_{AER}'.

Note that besides Q_{AER}, the candidate sets in Q_{DWU}, Q_{EMC}, Q_{EI} are also expended in a hierarchical manner for efficient implementation (see Section 4.1.1). Meanwhile, as the huge time cost of the EER estimation, we only employ the coarsest anchors as the candidates for Q_{EER}. We enlarge the numbers of anchor sets with the increase of datapoints, where the sizes of these anchor sets follow the proportion suggested in hierarchical anchor graph models [16], [23], [24].

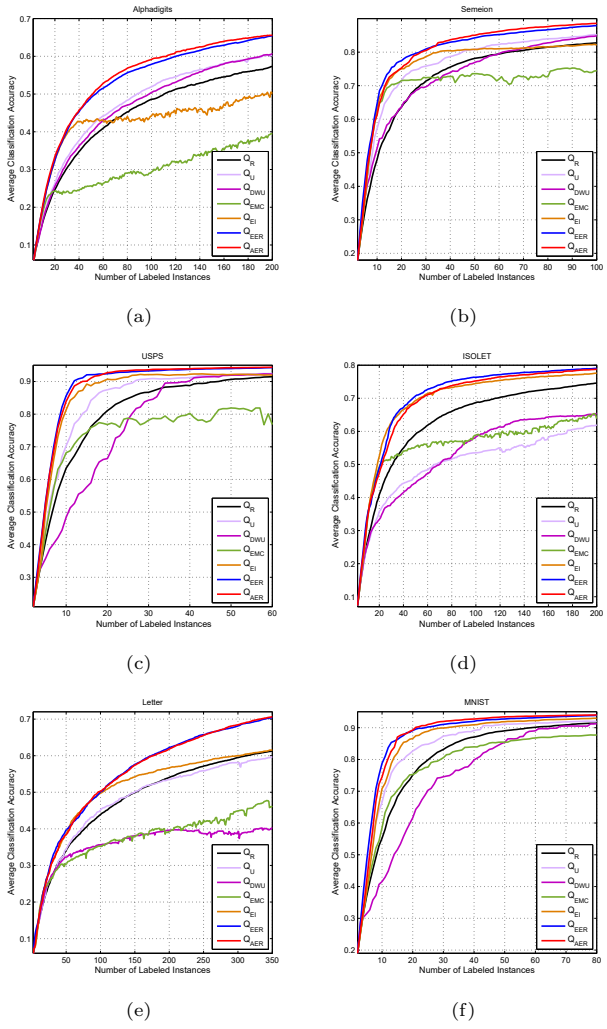


Figure 4: Average performance curves with respect to the number of labels on medium-size datasets.

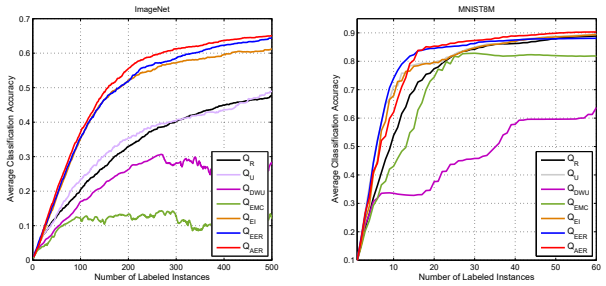
5.1.1 *On Medium-size Datasets*

For Alphadigits, Semeion, ISOLET, and Letter, we follow [24] and build 500-anchor-graphs. We build 2,000-500-anchor-graphs and 5,000-1,250-anchor-graphs for USPS and MNIST, respectively. We empirically set λ to 0.1. For active learning, only 2 instances are randomly sampled as the initial labeled data. Based on 20 trials, the average accuracy curves are displayed in Fig.4, and the time costs are listed in Table 4.

[2]available at http://www.cs.nyu.edu/~roweis/data.html
[3]available at http://archive.ics.uci.edu/ml
[4]available at http://www.cad.zju.edu.cn/home/dengcai/
[5]available at http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets
[6]available at http://yann.lecun.com/exdb/mnist
[7]We randomly select 200 classes from ImageNet [19] and build a subset with 256,091 images. We extract their 4,096-D CNN features via AlexNet [11] and perform PCA to reduce the dimension to 512.

Table 4: Comparison of average time costs (in seconds per query) on medium-size datasets.

Dataset	Q_U	Q_{DWU}	Q_{EMC}	Q_{EI}	Q_{EER}	Q_{AER}
Alphadigits	0.01	0.03	0.04	0.04	1.80	0.04
Semeion	0.01	0.01	0.02	0.02	0.67	0.02
USPS	0.01	0.02	0.02	0.02	2.45	0.02
ISOLET	0.02	0.04	0.03	0.04	7.94	0.05
Letter	0.03	0.10	0.11	0.17	13.81	0.20
MNIST	0.04	0.18	0.15	0.17	30.16	0.21



(a) (b)

Figure 5: Average accuracy curves with respect to the number of labels on large-size datasets.

Table 5: Comparison of average time costs (in seconds per query) on large-size datasets.

Dataset	Q_U	Q_{DWU}	Q_{EMC}	Q_{EI}	Q_{EER}	Q_{AER}
ImageNet	2.15	10.05	4.19	7.61	90.73	9.35
MNIST8M	4.22	14.47	10.97	12.19	198.73	16.15

From these results, we can obtain the following observations. Firstly, compared with Q_R, Q_U obtains higher accuracies on Semeion, USPS, and MNIST, and receives comparable or even worse performance on Alphadigits, ISOLET, and Letter. The reason behind is that, the uncertainty-based sampling is insufficiently effective to discover queries that actually belong to new classes, especially when the number of classes is large. Secondly, Q_{DWU} performs worse than Q_U in most cases, which indicates that directly combing the density and the entropy is not generally suitable for all classifiers. Thirdly, as Q_{EMC} only considers the changes on the parameters of the classifier, it obtains worse performance than Q_R with the increase of labels. In contrast, by introducing the data distribution, Q_{EI} can achieve much higher accuracies, which is consistent with the theoretical analysis in Section 4.4.2. Fourthly, Q_{AER} obtains comparable or even better performance than Q_{EER}, and its performance is more consistent to that of Q_{EER} than others. This result empirically demonstrates the effectiveness of our AER criterion on the error reduction estimation. Finally, when considering both the efficiency and the effectiveness, Q_{AER} highlights its strengths over other compared methods.

5.1.2 *On Large-size Datasets*

We also conduct experiments on ImageNet and MNIST8M with 100,000-10,000-2,500-anchor-graphs. We empirically set λ to 0.01 in HAGR. By repeating the similar process, we report the average classification accuracies over 10 trials in Fig.5 and list the time costs in Table 5.

From these results, the following observations can be made. Firstly, similar to the pervious results, Q_{EI} outperforms Q_{EMC} by giving consideration to data distributions. Secondly, compared with Q_{EI} and Q_{EER}, Q_{AER} can obtain higher accuracies after a few iterations of query selection. It means that introducing uncertainty into active learning criteria will be beneficial to tune decision boundaries. Thirdly, when taking account into the time cost in Table 5, we further confirm the superior performance of Q_{AER} for scalable active learning.

5.2 Effectiveness Analysis

5.2.1 *On the Formulation of AER*

In AER, ϵ is fixed to $\bar{\mathcal{E}}$ to control the confidence of the approximated ratio. To demonstrate the effectiveness, we compare Q_{AER} with the following two intermediate versions:

(1). $Q_{AER,Ada}$: This method follows [14] and sets ϵ to different values, e.g., $\{0, 0.1, \ldots, 1\}$ to obtain several sub-queries, and then refines the best query from them based on EER.

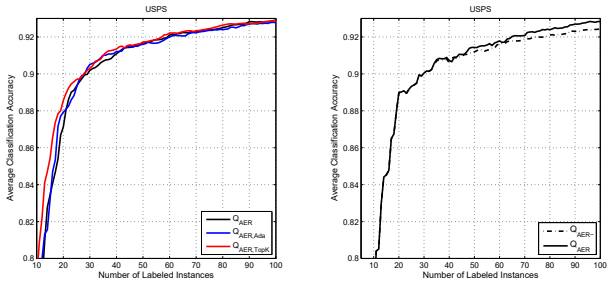
(2). $Q_{AER,TopK}$: This method first selects top $K(K{=}10)$ datapoints based on AER as sub-queries and then refines the best query from them based on EER.

We conduct experiments on the USPS dataset with 500-125-anchor-graphs. The average accuracy curves over 20 trias of these three methods are displayed in Fig.6(a).

From this figure, we observe that Q_{AER} can obtain comparable performance to $Q_{AER,Ada}$, which requires the extra time cost of query refining. The reason behind is that, although the setting of ϵ in $Q_{AER,Ada}$ is more flexible, most of them are useless. For example, at initial stages, the sub-queries with small expected impact values over all datapoints will not lead to rapid improvements. With the error reducing, the sub-queries with large impact values may not improve decision boundaries. In contrast, Q_{AER} prefers the datapoints with large impact values at first and those near decision boundaries with the increase of labels. That is, AER adaptively weighs the impact over all datapoints against the accumulated uncertainty over nearby datapoints. Meanwhile, we observe that although Q_{AER} performs slightly worse than $Q_{AER,TopK}$ at the early stages, it obtains comparable accuracies with the increase of labels. This result also implies that, we may further improve the performance of Q_{AER} with an extra query refining step, where the time cost is still much smaller than that of the direct EER-based query selection.

5.2.2 *On the Hierarchical Candidate Expansion*

We finally investigate the effectiveness of the candidate expansion based on hierarchical anchor graphs. We compare Q_{AER} with Q_{AER^-}, which is a simplified Q_{AER} without the candidate expansion via Eq.8. The average accuracy curves of these two methods are shown in Fig.6(c).



(a) (b)

Figure 6: Average accuracy curves with respect to the number of labeled instances on USPS.

From this figure, we observe that the accuracies of $\mathrm{Q_{AER}}$ and $\mathrm{Q_{AER}}$- are similar at initial stages. It means that, although $\mathrm{Q_{AER}}$ expands the candidate set from the beginning, it will not immediately fall into finer candidates for local accuracy improvements. Once the performance becomes better with the increase of labeled data, $\mathrm{Q_{AER}}$ will pay attention to these finer candidates and bring higher classification accuracies. This result further demonstrates the effectiveness of introducing finer candidates for tuning decision boundaries.

6 CONCLUSIONS

This paper proposed a novel query selection criterion called approximated error reduction (AER). Different from other criteria, AER estimates the error reduction of a datapoint based on its expected impact over all datapoints and its approximated ratio between the error reduction and the impact over its nearby datapoints. Meanwhile, AER employs hierarchical anchor graphs to expand a small candidate set with the increase of labels, which further accelerates its estimation. Benefiting from AER, we can obtain an efficient estimation of the error reduction without incrementally re-inferring labels of massive datapoints. We introduced AER into an efficient semi-supervised classifier for scalable active learning. The experiments on publicly available datasets demonstrated both the efficiency and the effectiveness of our approach.

It is worthwhile to note that since the supervised and semi-supervised classifiers in the literature [8], [22], [24], [28] have similar solutions to HAGR in their model optimization, our future work includes the integration of AER with them, where the proposed fast impact estimation can be generalized. In these works, all AER-based approaches are supposed to be much fater than EER, as the repeatedly label re-inference is not required in AER.

APPENDIXES

A. Proof of Proposition 1

Let $f(\alpha,\beta)$ denote $f_1(\alpha)\times f_2(\beta)$, where $f_1(\alpha)=\alpha$ and $f_2(\beta)=(\beta)^{1-\epsilon}$ are two mapping functions on the variables α, $\beta>0$ with $\epsilon\in(0,1)$, respectively. Denote $r_\alpha=\frac{\alpha_1}{\alpha_2}$, and $r'_\alpha=\frac{f_1(\alpha_1)}{f_1(\alpha_2)}$ as the original ratio and the mapped ratio on α, respectively, and $r_\beta=\frac{\beta_1}{\beta_2}$, and $r'_\beta=\frac{f_2(\beta_1)}{f_2(\beta_2)}$ as the original ratio and the mapped ratio on β, respectively. We have (1). $r_\beta>r'_\beta>1(r_\beta<r'_\beta<1)$ if $r_\beta>1(r_\beta<1)$. (2). $r'_\alpha=r_\alpha$. That is, the mapped r'_β is closer to 1 than r_β, and the mapped r'_α is same to r_α, namely, the difference of two βs corresponding to two instances is reduced via the mapping function f_2, and that of two αs is kept via f_1. As a consequence, it leads to the reduction of the influence of β on the objective value, and relatively increases the influence of α. According to Eq.6, let α denote the expected impact $\mathcal{I}_q$, and β be the approximated ratio $\frac{\mathcal{E}r_{\langle q\rangle}}{\mathcal{I}_{\langle q\rangle}}$, which completes the proof.

B. Fast Impact Estimation

Below we presents the derivation of the fast impact estimation used in Section 4.1.3.

Let $\mathbf{z}_q^{\mathrm{H}}$ be the cascaded inter-set adjacency vector of $\mathbf{x}_q$, and $\hat{\mathbf{y}}_{qr}$ be its class indicator vector where only the r-th element is 1. To obtain Eq.10, the traces of the following terms are needed: ${\mathbf{A}^{+\hat{y}_{qr}}}^{\mathrm{T}}\mathbf{\Delta}\mathbf{A}^{+\hat{y}_{qr}}$, ${\mathbf{A}^{+\hat{y}_{qr}}}^{\mathrm{T}}\mathbf{\Delta}\mathbf{A}$, and $\mathbf{A}^{\mathrm{T}}\mathbf{\Delta}\mathbf{A}$, where

$$\mathbf{A}^{+\hat{y}_{qr}}=\left({\mathbf{z}_q^{\mathrm{H}}}^{\mathrm{T}}\mathbf{z}_q^{\mathrm{H}}+{\mathbf{Z}_{\mathrm{L}}^{\mathrm{H}}}^{\mathrm{T}}\mathbf{Z}_{\mathrm{L}}^{\mathrm{H}}+\lambda\mathbf{r}\mathbf{L}\right)^{-1}\left({\mathbf{z}_q^{\mathrm{H}}}^{\mathrm{T}}\hat{\mathbf{y}}_{qr}+{\mathbf{Z}_{\mathrm{L}}^{\mathrm{H}}}^{\mathrm{T}}\mathbf{Y}_{\mathrm{L}}\right).$$

Suppose $\mathbf{M}=({\mathbf{Z}_{\mathrm{L}}^{\mathrm{H}}}^{\mathrm{T}}\mathbf{Z}_{\mathrm{L}}^{\mathrm{H}}+\lambda\mathbf{r}\mathbf{L})$, then $\mathbf{A}^{+\hat{y}_{qr}}=({\mathbf{z}_q^{\mathrm{H}}}^{\mathrm{T}}\mathbf{z}_q^{\mathrm{H}}+\mathbf{M})^{-1}$ $({\mathbf{z}_q^{\mathrm{H}}}^{\mathrm{T}}\hat{\mathbf{y}}_{qr}+{\mathbf{Z}_{\mathrm{L}}^{\mathrm{H}}}^{\mathrm{T}}\mathbf{Y}_{\mathrm{L}})$. By matrix inversion, we can therefore calculate $\mathbf{A}^{+\hat{y}_{qr}}$ as

$$(\mathbf{M}^{-1}-\frac{\mathbf{M}^{-1}\mathbf{z}_q^{\mathrm{H}}{\mathbf{z}_q^{\mathrm{H}}}^{\mathrm{T}}\mathbf{M}^{-1}}{{\mathbf{z}_q^{\mathrm{H}}}^{\mathrm{T}}\mathbf{M}^{-1}\mathbf{z}_q^{\mathrm{H}}+1})({\mathbf{z}_q^{\mathrm{H}}}^{\mathrm{T}}\hat{\mathbf{y}}_{qr}+{\mathbf{Z}_{\mathrm{L}}^{\mathrm{H}}}^{\mathrm{T}}\mathbf{Y}_{\mathrm{L}})$$

Let $\hat{\mathbf{M}}=\mathbf{M}^{-1}$, $\alpha_q=\mathbf{z}_q^{\mathrm{H}}\hat{\mathbf{M}}{\mathbf{z}_q^{\mathrm{H}}}^{\mathrm{T}}$, and $\beta_q=\frac{1}{1+\alpha_q}$. We can substitute the above results into the trace of the first term $({\mathbf{A}^{+\hat{y}_{qr}}}^{\mathrm{T}}\mathbf{\Delta}\mathbf{A}^{+\hat{y}_{qr}})$ and obtain it:

$$\begin{aligned}
&\mathrm{trace}[(\hat{\mathbf{y}}_{qr}^{\mathrm{T}}\mathbf{z}_q^{\mathrm{H}}+{\mathbf{Y}_{\mathrm{L}}}^{\mathrm{T}}\mathbf{Z}_{\mathrm{L}}^{\mathrm{H}})(\mathbf{I}-\beta_q\hat{\mathbf{M}}\mathbf{z}_q^{\mathrm{H}}{\mathbf{z}_q^{\mathrm{H}}}^{\mathrm{T}})\hat{\mathbf{M}}\mathbf{\Delta}\\
&\hat{\mathbf{M}}(\mathbf{I}-\beta_q\mathbf{z}_q^{\mathrm{H}}{\mathbf{z}_q^{\mathrm{H}}}^{\mathrm{T}}\hat{\mathbf{M}})({\mathbf{z}_q^{\mathrm{H}}}^{\mathrm{T}}\hat{\mathbf{y}}_{qr}+{\mathbf{Z}_{\mathrm{L}}^{\mathrm{H}}}^{\mathrm{T}}\mathbf{Y}_{\mathrm{L}})]\\
=&\mathrm{trace}[\hat{\mathbf{y}}_{qr}^{\mathrm{T}}(\mathbf{z}_q^{\mathrm{H}}\hat{\mathbf{M}}\mathbf{\Delta}\hat{\mathbf{M}}{\mathbf{z}_q^{\mathrm{H}}}^{\mathrm{T}})\hat{\mathbf{y}}_{qr}]+2\mathrm{trace}[\hat{\mathbf{y}}_{qr}^{\mathrm{T}}(\mathbf{z}_q^{\mathrm{H}}\hat{\mathbf{M}}\mathbf{\Delta}\hat{\mathbf{M}}{\mathbf{Z}_{\mathrm{L}}^{\mathrm{H}}}^{\mathrm{T}}\mathbf{Y}_{\mathrm{L}})]\\
&-2\beta_q\mathrm{trace}[\hat{\mathbf{y}}_{qr}^{\mathrm{T}}(\mathbf{z}_q^{\mathrm{H}}\hat{\mathbf{M}}\mathbf{\Delta}\hat{\mathbf{M}}{\mathbf{z}_q^{\mathrm{H}}}^{\mathrm{T}})(\mathbf{z}_q^{\mathrm{H}}\mathbf{M}{\mathbf{z}_q^{\mathrm{H}}}^{\mathrm{T}})\hat{\mathbf{y}}_{qr}]\\
&-2\beta_q\mathrm{trace}[\hat{\mathbf{y}}_{qr}^{\mathrm{T}}(\mathbf{z}_q^{\mathrm{H}}\hat{\mathbf{M}}\mathbf{\Delta}\hat{\mathbf{M}}{\mathbf{z}_q^{\mathrm{H}}}^{\mathrm{T}})(\mathbf{z}_q^{\mathrm{H}}\mathbf{M}{\mathbf{Z}_{\mathrm{L}}^{\mathrm{H}}}^{\mathrm{T}}\mathbf{Y}_{\mathrm{L}})]\\
&-2\beta_q\mathrm{trace}[\hat{\mathbf{y}}_{qr}^{\mathrm{T}}(\mathbf{z}_q^{\mathrm{H}}\hat{\mathbf{M}}{\mathbf{z}_q^{\mathrm{H}}}^{\mathrm{T}})(\mathbf{z}_q^{\mathrm{H}}\hat{\mathbf{M}}\mathbf{\Delta}\hat{\mathbf{M}}{\mathbf{Z}_{\mathrm{L}}^{\mathrm{H}}}^{\mathrm{T}}\mathbf{Y}_{\mathrm{L}})]\\
&+\beta_q^2\mathrm{trace}[(\hat{\mathbf{y}}_{qr}^{\mathrm{T}}(\mathbf{z}_q^{\mathrm{H}}\hat{\mathbf{M}}{\mathbf{z}_q^{\mathrm{H}}}^{\mathrm{T}})(\mathbf{z}_q^{\mathrm{H}}\hat{\mathbf{M}}\mathbf{\Delta}\hat{\mathbf{M}}{\mathbf{z}_q^{\mathrm{H}}}^{\mathrm{T}})(\mathbf{z}_q^{\mathrm{H}}\mathbf{M}{\mathbf{z}_q^{\mathrm{H}}}^{\mathrm{T}})\hat{\mathbf{y}}_{qr}]\\
&+2\beta_q^2\mathrm{trace}[\hat{\mathbf{y}}_{qr}^{\mathrm{T}}(\mathbf{z}_q^{\mathrm{H}}\hat{\mathbf{M}}{\mathbf{z}_q^{\mathrm{H}}}^{\mathrm{T}})(\mathbf{z}_q^{\mathrm{H}}\hat{\mathbf{M}}\mathbf{\Delta}\hat{\mathbf{M}}{\mathbf{z}_q^{\mathrm{H}}}^{\mathrm{T}})(\mathbf{z}_q^{\mathrm{H}}\mathbf{M}{\mathbf{Z}_{\mathrm{L}}^{\mathrm{H}}}^{\mathrm{T}}\mathbf{Y}_{\mathrm{L}})]\\
&+\mathrm{trace}[(\mathbf{Y}_{\mathrm{L}}^{\mathrm{T}}\mathbf{Z}_{\mathrm{L}}^{\mathrm{H}}\hat{\mathbf{M}})\mathbf{\Delta}(\hat{\mathbf{M}}{\mathbf{Z}_{\mathrm{L}}^{\mathrm{H}}}^{\mathrm{T}}\mathbf{Y}_{\mathrm{L}})]\\
&-2\beta_q\mathrm{trace}[(\mathbf{Y}_{\mathrm{L}}^{\mathrm{T}}\mathbf{Z}_{\mathrm{L}}^{\mathrm{H}}\hat{\mathbf{M}}\mathbf{\Delta}\hat{\mathbf{M}}{\mathbf{z}_q^{\mathrm{H}}}^{\mathrm{T}})(\mathbf{z}_q^{\mathrm{H}}\mathbf{M}{\mathbf{Z}_{\mathrm{L}}^{\mathrm{H}}}^{\mathrm{T}}\mathbf{Y}_{\mathrm{L}})]\\
&+\beta_q^2\mathrm{trace}[(\mathbf{Y}_{\mathrm{L}}^{\mathrm{T}}\mathbf{Z}_{\mathrm{L}}^{\mathrm{H}}\hat{\mathbf{M}}{\mathbf{z}_q^{\mathrm{H}}}^{\mathrm{T}})(\mathbf{z}_q^{\mathrm{H}}\hat{\mathbf{M}}\mathbf{\Delta}\hat{\mathbf{M}}{\mathbf{z}_q^{\mathrm{H}}}^{\mathrm{T}})(\mathbf{z}_q^{\mathrm{H}}\hat{\mathbf{M}}{\mathbf{Z}_{\mathrm{L}}^{\mathrm{H}}}^{\mathrm{T}}\mathbf{Y}_{\mathrm{L}})].
\end{aligned}$$

Suppose $\gamma_q=\mathbf{z}_q^{\mathrm{H}}(\hat{\mathbf{M}}\mathbf{\Delta}\hat{\mathbf{M}}){\mathbf{z}_q^{\mathrm{H}}}^{\mathrm{T}}$, $\boldsymbol{\varphi}_q=\mathbf{z}_q^{\mathrm{H}}\mathbf{A}$, $\boldsymbol{\phi}_q=\mathbf{z}_q^{\mathrm{H}}\hat{\mathbf{M}}\mathbf{\Delta}\mathbf{A}$, and $\delta=\mathrm{trace}(\mathbf{A}^{\mathrm{T}}\mathbf{\Delta}\mathbf{A})$, where $\mathbf{A}=\hat{\mathbf{M}}{\mathbf{Z}_{\mathrm{L}}^{\mathrm{H}}}^{\mathrm{T}}\mathbf{Y}_{\mathrm{L}}$. Besides, we obtain $\mathrm{trace}(\hat{\mathbf{y}}_{qr}^{\mathrm{T}}\gamma_q\hat{\mathbf{y}}_{qr})=\gamma_q$, and $\mathrm{trace}(\hat{\mathbf{y}}_{qr}^{\mathrm{T}}\boldsymbol{\phi}_q)=(\boldsymbol{\phi}_q)_r$, where $(\cdot)_r$ is the r-th element of the inside vector. By substituting them

into the above equation, we obtain the trace of the first term:

$$\delta + (1 - 2\alpha_q\beta_q + \alpha_q^2\beta_q^2)\gamma_q + (2\alpha_q\beta_q^2\gamma_q - 2\beta_q\gamma_q)(\boldsymbol{\varphi}_q)_r + (2 - 2\alpha_q\beta_q)(\boldsymbol{\phi}_q)_r - 2\beta_q\boldsymbol{\phi}_q{}^{\mathrm{T}}\boldsymbol{\varphi}_q + \beta_q^2\gamma_q\boldsymbol{\varphi}_q{}^{\mathrm{T}}\boldsymbol{\varphi}_q,$$

Similarly, the trace of the second term can be obtained:

$$\begin{aligned}
&\mathrm{trace}[\mathbf{A}\boldsymbol{\Delta}\hat{\mathbf{M}}(\mathbf{I} - \beta_q\mathbf{z}_q^{\mathrm{H}}\mathbf{z}_q^{\mathrm{H}\mathrm{T}}\hat{\mathbf{M}})(\mathbf{z}_q^{\mathrm{H}\mathrm{T}}\hat{\mathbf{y}}_{qr} + \mathbf{Z}_{\mathrm{L}}^{\mathrm{H}\mathrm{T}}\mathbf{Y}_{\mathrm{L}})] \\
=&\mathrm{trace}[(\mathbf{A}\boldsymbol{\Delta}\hat{\mathbf{M}}\mathbf{z}_q^{\mathrm{H}\mathrm{T}})\hat{\mathbf{y}}_{qr}] + \mathrm{trace}(\mathbf{A}\boldsymbol{\Delta}\mathbf{A}) \\
&- \mathrm{trace}[\beta_q(\mathbf{A}\boldsymbol{\Delta}\hat{\mathbf{M}}\mathbf{z}_q^{\mathrm{H}})(\mathbf{z}_q^{\mathrm{H}\mathrm{T}}\hat{\mathbf{M}}\mathbf{z}_q^{\mathrm{H}\mathrm{T}})\hat{\mathbf{y}}_{qr}] \\
&- \mathrm{trace}[\beta_q(\mathbf{A}\boldsymbol{\Delta}\hat{\mathbf{M}}\mathbf{z}_q^{\mathrm{H}})(\mathbf{z}_q^{\mathrm{H}\mathrm{T}}\hat{\mathbf{M}}\mathbf{Z}_{\mathrm{L}}^{\mathrm{H}\mathrm{T}}\hat{\mathbf{y}}_{\mathrm{L}})] \\
=&\delta + (1 - \alpha_q\beta_q)(\boldsymbol{\phi}_q)_r - \beta_q\boldsymbol{\varphi}_q{}^{\mathrm{T}}\boldsymbol{\phi}_q.
\end{aligned}$$

We briefly analyze the above time cost. First, for N_q candidates, the total time cost of α_qs is $O(N_h^2N_q)$, and the following cost for β_q is $O(N_q)$. Then for these candidates, the time cost of γ_qs is $O(N_h^3+N_h^2N_q)$, and that of all $\boldsymbol{\phi}_q$s, $\boldsymbol{\varphi}_q$s, $\boldsymbol{\phi}_q{}^{\mathrm{T}}\boldsymbol{\varphi}_q$s and $\boldsymbol{\varphi}_q{}^{\mathrm{T}}\boldsymbol{\varphi}_q$s is $O(N_hN_qC+N_h^2C+N_qC^2)$. Therefore, for N_q candidates, the time cost of their expected impact estimation is $O(N_h^2N_q+N_h^3+N_hN_qC+N_h^2C+N_qC^2)$. Since $N_q \geq N_h \gg C$, it can be simplified as $O(N_h^2N_q)$. Of note, these impact values can be computed via direct matrix operations rather than multiple iterations.

ACKNOWLEDGMENTS

This work was partially supported by the National Key Research and Development Program of China under grant 2016YFB1000901, the National Nature Science Foundation of China under grant 61632007, 61772171, and 71490725, 91746209, and the Program for Changjiang Scholars and Innovative Research Team in University of the Ministry of Education, China under grant IRT17R32.

REFERENCES

[1] Oisin Mac Aodha, Neill D. F. Campbell, Jan Kautz, and Gabriel J. Brostow. 2014. Hierarchical subquery evaluation for active learning on a graph. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*. 564–571.

[2] Deng Cai and Xiaofei He. 2012. Manifold adaptive experimental design for text categorization. *IEEE Transactions on Knowledge and Data Engineering* 24, 4 (2012), 707–719.

[3] Wenbin Cai, Ya Zhang, and Jun Zhou. 2013. Maximizing expected model change for active learning in regression. In *Proceedings of the IEEE International Conference on Data Mining*. 51–60.

[4] Xiaojun Chang, Yao-Liang Yu, and Yi Yang. 2017. Robust Top-k Multiclass SVM for Visual Category Recognition. In *Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 75–83.

[5] Wei-Lin Chiang, Mu-Chu Lee, and Chih-Jen Lin. 2016. Parallel Dual Coordinate Descent Method for Large-scale Linear Classification in Multi-core Environments. In *Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1485–1494.

[6] Jan K Chorowski, Dzmitry Bahdanau, Dmitriy Serdyuk, Kyunghyun Cho, and Yoshua Bengio. 2015. Attention-based models for speech recognition. In *Proceedings of the Advances in Neural Information Processing Systems*. 577–585.

[7] Gautam Dasarathy, Robert Nowak, and Xiaojin Zhu. 2015. S2: An efficient graph based active learning algorithm with application to nonparametric classification. In *Proceedings of the Annual Conference on Learning Theory*. 503–522.

[8] Weijie Fu, Meng Wang, Shijie Hao, and Tingting Mu. 2017. FLAG: faster learning on anchor graph with label predictor optimization. *IEEE Transactions on Big Data* (2017). To appear.

[9] Sheng-Jun Huang, Rong Jin, and Zhi-Hua Zhou. 2014. Active learning by querying informative and representative examples. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 10, 36 (2014), 1936–1949.

[10] Ajay J Joshi, Fatih Porikli, and Nikolaos P Papanikolopoulos. 2012. Scalable active learning for multiclass image classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34, 11 (2012), 2259–2273.

[11] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Proceedings of the Advances in Neural Information Processing Systems*. 1097–1105.

[12] David D Lewis and William A Gale. 1994. A sequential algorithm for training text classifiers. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*. 3–12.

[13] Xin Li and Yuhong Guo. 2013. Active Learning with Multi-Label SVM Classification. In *Proceedings of the International Joint Conferences on Artificial Intelligence*. 1479–1485.

[14] Xin Li and Yuhong Guo. 2013. Adaptive active learning for image classification. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*. 859–866.

[15] Christopher H Lin, M Mausam, and Daniel S Weld. 2016. Re-Active Learning: Active Learning with Relabeling. In *Proceedings of the AAAI Conference on Artificial Intelligence*. 1845–1852.

[16] Wei Liu, Junfeng He, and Shih Fu Chang. 2010. Large graph construction for scalable semi-supervised learning. In *Proceedings of the International Conference on Machine Learning*. 679–686.

[17] Marius Muja and David G Lowe. 2014. Scalable nearest neighbor algorithms for high dimensional data. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36, 11 (2014), 2227–2240.

[18] Carlos Riquelme, Mohammad Ghavamzadeh, and Alessandro Lazaric. 2017. Active learning for accurate estimation of linear models. In *Proceedings of the International Conference on Machine Learning*.

[19] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. 2015. ImageNet large scale visual recognition challenge. *International Journal of Computer Vision* 115, 3 (2015), 211–252.

[20] Burr Settles. 2010. Active learning literature survey. *University of Wisconsin, Madison* 52, 55-66 (2010), 11.

[21] Burr Settles and Mark Craven. 2008. An analysis of active learning strategies for sequence labeling tasks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 1070–1079.

[22] Vikas Sindhwani, Partha Niyogi, Mikhail Belkin, and Sathiya Keerthi. 2005. Linear manifold regularization for large scale semi-supervised learning. In *Proceedings of the International Conference on Machine Learning*, Vol. 28.

[23] Meng Wang, Weijie Fu, Shijie Hao, Hengchang Liu, and Xindong Wu. 2017. Learning on big graph: Label inference and regularization with anchor hierarchy. *IEEE Transactions on Knowledge and Data Engineering* 29, 5 (2017), 1101–1114.

[24] Meng Wang, Weijie Fu, Shijie Hao, Dacheng Tao, and Xindong Wu. 2016. Scalable semi-supervised learning by efficient anchor graph regularization. *IEEE Transactions on Knowledge and Data Engineering* 28, 7 (2016), 1864–1877.

[25] Zheng Wang and Jieping Ye. 2015. Querying discriminative and representative samples for batch mode active learning. *ACM Transactions on Knowledge Discovery from Data* 9, 3 (2015), 17.

[26] Kai Yu, Jinbo Bi, and Volker Tresp. 2006. Active learning via transductive experimental design. In *Proceedings of the International Conference on Machine Learning*. 1081–1088.

[27] Raphael Yuster and Uri Zwick. 2005. Fast sparse matrix multiplication. *ACM Transactions on Algorithms* 1, 1 (2005), 2–13.

[28] Kai Zhang, Liang Lan, James T Kwok, Slobodan Vucetic, and Bahram Parvin. 2015. Scaling up graph-based semisupervised learning via prototype vector machines. *IEEE Transactions on Neural Networks and Learning Systems* 26, 3 (2015), 444–457.

[29] Xiaojin Zhu, John Lafferty, and Zoubin Ghahramani. 2003. Combining active learning and semi-supervised learning using gaussian fields and harmonic functions. In *Proceedings of the International Conference on Machine Learning*, Vol. 3.