Exploiting Context for Robustness to Label Noise in Active Learning

Sudipta Paul, *Member, IEEE*, Shivkumar Chandrasekaran, B.S. Manjunath, *Fellow, IEEE* and
Amit K. Roy-Chowdhury, *Fellow, IEEE*

Abstract—Several works in computer vision have demonstrated the effectiveness of active learning for adapting the recognition model when new unlabeled data becomes available. Most of these works consider that labels obtained from the annotator are correct. However, in a practical scenario, as the quality of the labels depends on the annotator, some of the labels might be wrong, which results in degraded recognition performance. In this paper, we address the problems of i) how a system can identify which of the queried labels are wrong and ii) how a multi-class active learning system can be adapted to minimize the negative impact of label noise. Towards solving the problems, we propose a noisy label filtering based learning approach where the inter-relationship (context) that is quite common in natural data is utilized to detect the wrong labels. We construct a graphical representation of the unlabeled data to encode these relationships and obtain new beliefs on the graph when noisy labels are available. Comparing the new beliefs with the prior relational information, we generate a dissimilarity score to detect the incorrect labels and update the recognition model with correct labels which result in better recognition performance. This is demonstrated in three different applications: scene classification, activity classification, and document classification.

Index Terms—Context, Label noise, Active learning.

I. INTRODUCTION

Most of the current visual recognition tasks are performed by supervised learning approaches, which require a lot of training data. Every day a lot of visual and text data is generated from various sources which we can manually label and utilize to update the recognition system. But manually labeling a huge amount of data is tedious work and it becomes expensive if human experts are used. To reduce the labeling task, one effective approach is to actively select informative samples for manual labeling and update the recognition models with these selected samples. This scheme is known as active learning. As all the training samples may not be useful for developing a recognition system, active learning can reduce the labeling cost without compromising the recognition performance much [1], [2], [3], [4], [5], [6], [7], [8].

In most of the active learning works, it is assumed that the labels provided by the human labelers are correct. However, in a practical scenario, labels queried from non-expert labelers are prone to error due to perception variations or incorrect annotation [9]. Incorrect labels can adversely impact the

• Sudipta Paul, and Amit K. Roy-Chowdhury are with the Department of Electrical and Computer Engineering, University of California, Riverside, CA, USA. Shivkumar Chandrasekaran and B.S. Manjunath are with the Department of Electrical and Computer Engineering, University of California, Santa Barbara, CA, USA. E-mails: (spaul007@ucr.edu, shiv@ucsb.edu, manj@ucsb.edu, amitrc@ece.ucr.edu)

classification performance of the influenced classifier [10]. This adverse impact becomes severe in an active learning process as the amount of labeled data is limited. There are some works [11], [12], [13], [14] that consider active learning where an annotator can provide wrong labels. Most of these works [11], [12], [13] only consider label noise problem for binary classification. In contrast, visual recognition systems typically require to perform multi-class classification tasks. In [14], the authors studied multi-class multi-annotator active learning in the presence of label noise. They propose active learning with Robust Gaussian Process (RGP). However, as the computational cost of inference in Gaussian Process is $\mathcal{O}(n^3)$ [15], this approach is not applicable to large scale datasets. Moreover, there can be many applications where it may not be possible to get multiple annotators, e.g., those that require a high level of domain knowledge. Hence, the problem of multi-class active learning in the presence of label noise requires more exploration. Furthermore, none of the previous approaches consider detecting which of the queried labels are wrong. Detection of wrong labels is of significant importance as it can be valuable for many applications like dataset creation with minimal human effort, annotator expertise estimation, and identifying samples that are difficult to annotate.

In this work, we propose a Context-aware Noisy Label Detection (CNLD) approach to detect wrong labels and utilize CNLD to formalize an active learning framework to handle the adverse impact of label noise. In many applications, several works have shown how to utilize the relationships between data points, i.e., structure in the data, for different purposes. For example, the relationship is used to improve recognition performance in activity recognition [16][17], object recognition [18][19], and text classification [20] [21]. This inter-relationship is often termed as context. We also utilize the inter-relationships that are quite common in natural data to detect noisy labels. Generally, an incremental learning scenario that uses active selection has one or more initial seed models (classification model, relationship model) that are built on correct labels. We leverage the seed relationship model to obtain prior relational information among the data classes. When new data with queried labels are available, we infer its relational information using graphical representations, compare it with prior relations, and based on that obtain a dissimilarity score which is a notion of how likely an instance is incorrectly labeled. Utilizing this estimation, we detect which labels are wrong, filter out the wrong labels, and continue the learning process with correct labels. The motivation for this noisy label detection approach is that an instance assigned with the wrong
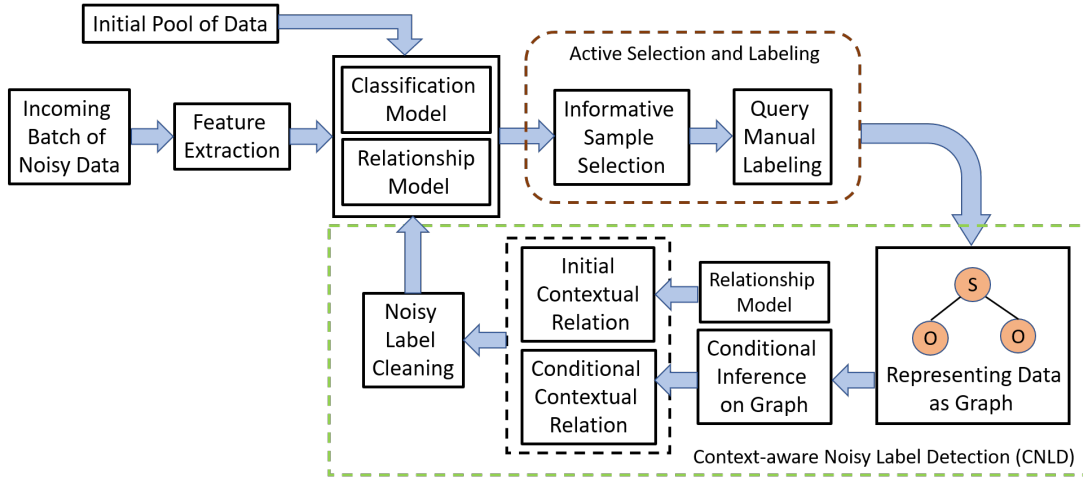
Fig. 1: Proposed framework for label noise-robust active learning scheme. Initial classification model $\mathcal{M}^{t_0}$ and relationship model $\mathcal{R}^{t_0}$ are obtained using initial correctly labeled pool of data. When a new batch of data is available, it selects an informative subset of samples and queries for human labeling where a fraction of the queried labels is considered wrong. Graphical representations are constructed to encode contextual information. Conditional inference on the graph gives new edge beliefs which are compared with prior beliefs to obtain a measure of how likely a label is wrong. Then the classification model $\mathcal{M}^{t_0}$ and relationship model $\mathcal{R}^{t_0}$ are updated using only the correct labels.

label will lead to relational information among data classes that are not consistent with the known prior relational information among data classes.

Framework Overview. Figure 1 illustrates the framework of our proposed approach. The framework is based on two assumptions: i) there is an underlying structure in the data which provides contextual relationships among the data classes and ii) we have an initial pool of data that is correctly labeled. Both of these are weak assumptions as almost all visual data occurring naturally is structured and most active learning methods starts with an initial seed model which is learned using a small set of labels queried from expert annotators. The method starts with training a classification model $\mathcal{M}$ and a relationship model $\mathcal{R}$ using the initial pool of correctly labeled data. When a new batch of unlabeled data is available, an informative subset of the samples is actively selected. These informative samples are queried for human labeling. We consider the practical scenario where a subset of the queried labels is wrong. Our goal is to improve the recognition performance by updating the classifier with queried labels.

Since the incorrect labels may confuse the learning process resulting in poor performance [22], we formulate a noisy label filtering based approach to reduce the influence of wrong labels during the learning process. To filter the noisy labels, we start by representing the queried instances along with their linked elements and defined attributes as graphs. Using the learned classifier, a probability mass function over the possible classes is obtained for each queried instance and its linked elements. We assign this probability mass function as node potentials. Edge potentials are obtained from the current relationship model. A message-passing algorithm is used to perform conditional inference. Conditional inference gives new edge beliefs which we consider as the posterior contextual information. We compare the posterior contextual

information with prior contextual relation (obtained from the current relationship model) to compute a dissimilarity measure which is used to detect wrong labels. Then the classification model $\mathcal{M}$ and the relationship model $\mathcal{R}$ are updated using the filtered labels.

Main Contributions. The main contributions of the work are as follows.

- We derive a dissimilarity score to determine wrong labels by exploiting the inter-relationship among data categories, which is ubiquitous in natural data.
- We formalize a general active learning framework that utilizes the inter-relationship based dissimilarity score to filter noisy labels provided by the annotator.
- We empirically evaluate the performance of the Context-aware Noisy Label Detection (CNLD) approach, as well as its positive impact on active learning, on three different applications.

II. RELATED WORKS

There has been a considerable amount of work on active learning. Most of the active learning algorithms use the uncertainty of the classifier as a measure of informativeness of an unlabeled data, e.g., entropy [3], best vs. second best [23], classifier margin [24]. Another common concept in active learning algorithms is expected model output change utilized in [25], [5], [26]. All of these methods consider the samples to be independent of each other. There have been some active learning methods that utilize relational information [27], [1], [28]. Recently some active learning methods are developed to scale well with deep learning network, e.g., core-set approach [29], deep Bayesian approach [30], learning loss based approach [31], variational adversarial approach [32].

There have been some works on active learning in the presence of label noise. Most of the works have a base

setting where the learner is given an input space $\mathcal{X}$, a label space $\mathcal{L}$, and a hypothesis class $\mathcal{H}$. The goal is to select a hypothesis from the hypothesis class $\mathcal{H}$ which is closest to the hypothesis that generates the ground truth labels. Two types of noisy label settings are commonly used in these works: i) random classification Noise (RCN), where each label is flipped with a probability that is independent of the instances, and ii) increase of noise rate near the decision boundary. In [33], the RCN setting is used and addressed by repeatedly querying an example. The sampling strategy in [34] utilizes Extrinsic Jensen-Shanon (EJS) divergence. In [35], [36], [37], [38], the second setting of noise where noise rate increases near decision boundary is studied. Works on agnostic active learning [39], [40], [37], [36], [41], [38] considers a fraction of label may disagree with the optimal hypothesis of the hypothesis class $\mathcal{H}$. However, maintaining a hypothesis class may not be feasible for many computer vision applications.

There are some works on noise-robust active learning in the presence of multiple annotators. In [11], active learning of kernel machine ensemble in collaborative labeling when labels might not be noise free is explored. In [12], [13], a noise resilient probabilistic model for active learning of a Gaussian process classifier from crowds is used. However, these approaches are designed for binary classification task. Long et al. [14] have studied multi-class multi-annotator active learning using robust Gaussian process for visual recognition. In contrast, our work focuses on the presence of noisy labels for each annotator. Moreover, we are utilizing interrelationship among data to detect the noisy labels and improve the robustness of the active learning framework.

III. METHODOLOGY

Problem Definition. Suppose, we have an initial set of data instances $\mathcal{L}$ that is correctly labeled. We extract features $\boldsymbol{X}^{\mathcal{L}}$ from this initial set of labeled data and train baseline classification model $\mathcal{M}$ and relationship model $\mathcal{R}$. Then a new batch of unlabeled data $\mathcal{U}$ consisting of N data instances becomes available. We represent the extracted features of the unlabeled set of data as $\{\boldsymbol{X}_j^{\mathcal{U}}\}_{j=1}^N$. To update the classification model $\mathcal{M}$, an active sample selection procedure selects a subset $\mathcal{Q}$ of k unlabelled data instances from the unlabeled set of data $\mathcal{U}$, where $k \leq N$. This set of k instances, $\mathcal{Q} = \{q_1, q_2, \ldots, q_k\}$ is queried for manual labeling. After labeling by human annotator, we have the obtained labels $Y' = \{y_1', y_2', \ldots, y_k'\}$. Considering the scenario where manual labeling is prone to error, we assume Ω fraction of the observed labels are wrong and the noise rate is unknown to the system. The true labels $Y = \{y_1, y_2, \ldots, y_k\}$ of selected set $\mathcal{Q}$ is also unknown. Our goal here is to identify and remove the wrong labels and update the classification model $\mathcal{M}$ and relationship model $\mathcal{R}$ with only the correct labels so that the wrong labels cannot influence the models adversely.

Noisy Label Generation. We consider two statistical models to generate noisy labels synthetically. The Noisy Completely at Random (NCAR) [9] statistical model is used to generate symmetric label noise and the Noisy at Random (NAR) [9] model is used to generate asymmetric label noise.
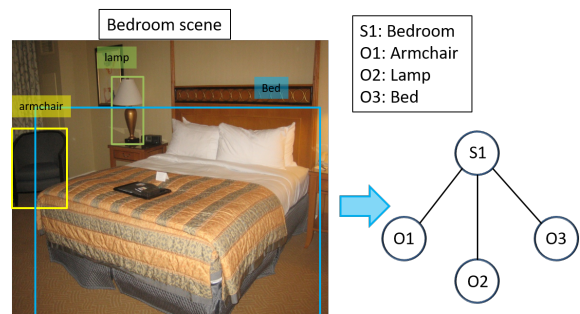

Fig. 2: An example illustration of how the instances are represented as graphical structure. In the scene classification task, the shown image has a scene tag: bedroom and 3 objects: bed, armchair, and lamp. We represent the image by a tree structured graph with four nodes (one scene node and three object nodes) and three edges (scene-object).

In the NCAR model, occurrence of an error has no relation with the instance or the label of that instance. Let the set of possible classes be $\mathcal{Z} = \{c_1, c_2, \ldots, c_n\}$ for a set of data with n classes and the selected noise rate is Ω for each class. Assigning equal noise rate Ω to each class results in symmetric noise. For the ith class, Ω fraction of randomly chosen labels are assigned with randomly chosen classes from the set $\mathcal{Z} \backslash \{c_i\}$ using the NCAR statistical model.

In the NAR model, occurrence of an error depends on the true label of the instance. We use this model to generate asymmetric noise as we can define which classes are more prone to label noise using the NAR model. Label noise generated by the NAR statistical model can be characterized by label transition matrix [9]. Let two random variables Y and $\tilde{Y}$ denotes the true label and the assigned label respectively. So the label transition matrix characterizing the label noise noise generation is,

$$\Lambda = \begin{pmatrix} P(\tilde{Y} = c_1 | Y = c_1) & \ldots & P(\tilde{Y} = c_n | Y = c_1) \\ \vdots & \ddots & \vdots \\ P(\tilde{Y} = c_1 | Y = c_n) & \ldots & P(\tilde{Y} = c_n | Y = c_n) \end{pmatrix} \quad (1)$$

Here, $P(\tilde{Y} = \tilde{y} | Y = y)$ is the label transition probabilities from true class to assign class. We use k-means clustering on the training data to obtain the transition probabilities, which is discussed in section (IV-D).

A. Modeling Contextual Relationships

Inspired by the success of graphical representation in encoding contextual relationships in several applications [42], [28], [18], we also utilize graphical representation to encode contextual relationships. Graph construction and how we define the node potentials and the edge potentials are described below.

Graph Formulation. We model the inter-relationship among the data by constructing an undirected graph $G = (V, E)$. Each node in V represents a single instance. The edges $E = \{(i, j) | v_i \text{ and } v_j \text{ are linked}\}$ represent the relationships between the data points. If related attributes $\mathcal{A}$ are present

in the structured data, we also model these into a graphical representation. Then the undirected graph $G = (V, E)$, modeling inter-relationship among the data points and also the relationships between the data points and related attributes $\mathcal{A}$ will have two types of node, $V = \{D, A\}$ and two types of edges, $E = \{D - D, D - A\}$. Here, D represents the set of nodes corresponding to the data instances and A represents the set of nodes corresponding to the related attributes. $D - D$ and $D - A$ are the relationships among data points and between data points and attributes respectively. Related attributes depend on the specific application, e.g., in scene classification, objects present in an image can be used as attributes. We consider the case of document classification where only the links between data points $(D - D)$ are present, scene classification where only the links between data points and related attributes $(D - A)$ are present and activity classification where both links between data points $(D - D)$ and links between data points and attributes $(D - A)$ are present. For example, in scene classification task, we represent data instance (scene) as scene node and detected objects as attribute nodes. As shown in Figure 2, we formulate a tree structured graph to represent the image.

Algorithm 1 Context-Aware Noisy Label Detection (CNLD)

Input:
- Initial correct classification model $\mathcal{M}^{t_0}$
- Initial correct relationship model $\mathcal{R}^{t_0}$
- Annotated noisy labels

Output: Detection of wrong labels
step 1: Calculate $P(\mathcal{C}^D|c)$ and $P(\mathcal{C}^A|c)$ from $\mathcal{R}^{t_0}$
for each element (q_i) of $\mathcal{Q}$ **do**
    step 2: Construct $G_i = (V_i, E_i)$
    step 3: Calculate $\hat{P}(\mathcal{C}_i^D|c)$ and $\hat{P}(\mathcal{C}_i^A|c)$ by conditional inference
    step 4: Calculate dissimilarity score l_i using Eqn. 7
end for
step 5: Estimate weight γ using Eqn. 8
step 6: Based on β detect noisy labels.

Node Potential. Let us consider that we have a classification task where the data belongs to one of n classes of $\{c_1, \ldots, c_n\}$. Given a classifier $\mathcal{M}$, it can generate probability estimate of a data instance belonging to any class of $\{c_1, \ldots, c_n\}$. The probability estimate of node j belonging to some class c_i can be expressed as $\mathcal{M}(\boldsymbol{X}_j, c_i)$. Consider an indicator function $\mathcal{I}(.)$ which takes as input a class c and provides as output a unit standard basis vector, i.e., $\mathcal{I}(c = c_1) = [1, 0, \ldots, 0]^T$. So the vector containing the node potentials of the jth node for n classes can be expressed as,

$$\varphi_j = \sum_{i=1}^{n} \mathcal{I}(c = c_i)\mathcal{M}(\boldsymbol{X}_j, c_i) \tag{2}$$

Edge Potential. The edge potentials are obtained using the co-occurrence frequency [19]. Co-occurrence statistics can give an estimate of how likely two data classes are related or how data classes are related to the attribute classes. For example, in an image tagged with 'bedroom' scene class,

objects such as 'bed' and 'chair' are more likely to occur than a 'car'. So the edge potentials represent the relationship weights among the data classes as well as between data classes and attribute classes. For two types of edges, we assign two different edge potential matrices $\boldsymbol{\Psi}_{D-D}$ and $\boldsymbol{\Psi}_{D-A}$. Here $\boldsymbol{\Psi}(i, j)$ is the co-occurrence frequency of class c_i with class c_j. Calculation of co-occurrence frequencies are application-specific and discussed in section IV.

B. Context-aware Noisy Label Detection

Suppose, a subset of data $\mathcal{Q} = \{q_1, q_2, \ldots, q_k\}$, consisting of k elements, is queried for human labeling. We consider that Ω fraction of the labels are incorrect. In this section, we discuss how we detect the incorrect labels using the graphical representations that encode the relationships among the data and also among data and attributes.

Contextual Relation. The relationship model $\mathcal{R}$ contains the co-occurrence information of different data classes and attribute classes. For a n class classification task with classes $\{c_1, c_2, \ldots, c_n\}$, we have $n \times n$ matrix $\boldsymbol{\Psi}_{D-D}$, where the $(i, j)^{th}$ value, $\boldsymbol{\Psi}_{D-D}(i, j)$ represents the co-occurrence statistics of data class c_i and data class c_j. Using this co-occurrence information we can calculate the probability of presence of ith data class c_i in presence of jth data class c_j by,

$$P(c_i|c_j) = \frac{\boldsymbol{\Psi}_{D-D}(j, i)}{\sum_{i=1}^{n} \boldsymbol{\Psi}_{D-D}(j, i)} \tag{3}$$

Similarly, from the relationship model $\mathcal{R}^{t_0}$, we have the co-occurrence frequency of data classes and attribute classes $\boldsymbol{\Psi}_{D-A}$. If there are m attribute classes $\{a_1, a_2, \ldots, a_m\}$, the probability of presence of ith attribute class a_i in presence of jth data class c_j can be expressed by,

$$P(a_i|c_j) = \frac{\boldsymbol{\Psi}_{D-A}(j, i)}{\sum_{i=1}^{n} \boldsymbol{\Psi}_{D-A}(j, i)} \tag{4}$$

Prior Relational Information. From the current relationship model $\mathcal{R}^{t_0}$, we know $\boldsymbol{\Psi}_{D-D}$ and $\boldsymbol{\Psi}_{D-A}$, which we consider as the prior edge beliefs. Suppose, in an n class classification problem with m classes of attributes, $\mathcal{C}^D$ is a random variable with sample space $\{c_1, c_2, \ldots, c_n\}$ and $\mathcal{C}^A$ is a random variable with sample space $\{a_1, a_2, \ldots, a_m\}$. Using the prior edge beliefs in Eqn 3 and 4, we obtain the conditional distribution $P(\mathcal{C}^D|c)$ and $P(\mathcal{C}^A|c)$, which we call the prior relational information for instances of $\mathcal{Q}$.

Dissimilarity Score Generation. In order to detect the wrong labels by exploiting the relationships, we construct graphical representation as described in III-A for each element of $\mathcal{Q}$ along with their linked data instances and attributes. Consider an instance q is linked with e data instances and attributes and the queried label is y'. We construct graph for q, which can be represented as $G = (V, E)$ where $V = \{v_1, v_2, \ldots, v_{e+1}\}$ and $E = \{(1, j)|j = 2, \ldots, e + 1\}$. Here the node v_1 represents the data instance q and nodes $v_2, \ldots, v_{m+1}$ are linked data instances and attributes of q. The graph has e edges that connects v_1 with all the other nodes. So G forms a tree structure. The node potentials φ and the edge

potentials $\boldsymbol{\Psi}_{D-D}$, $\boldsymbol{\Psi}_{D-A}$ are assigned using the classification model $\mathcal{M}^{t_0}$ and relationship model $\mathcal{R}^{t_0}$.

Now for each instance of $\mathcal{Q}$, we estimate its class conditional relatedness with other classes by making conditional inference on the representative graph. Conditional inference gives the pairwise conditional distribution of classes for each edge, which we call the posterior edge beliefs $\hat{\boldsymbol{\Psi}}$. Using the posterior edge beliefs of all edges in a graph, we estimate the posterior probability distribution conditioned on a class for each instance of $\mathcal{Q}$.

Suppose, we assign j^{th} class c_j to the instance q and make conditional inference on graph G. Let the number of $D-D$ edges is e^1 and the number of $D-A$ edges is e^2 in graph G. We estimate the posterior probabilities by,

$$\hat{P}(c_i|c_j) = \frac{1}{e^1} \frac{\sum_{k=1}^{e^1} \hat{\boldsymbol{\Psi}}_{(D-D)_k}(j,i)}{\sum_{k=1}^{e^1} \sum_{i=1}^{n} \hat{\boldsymbol{\Psi}}_{(D-D)_k}(j,i)}, \quad (5)$$

$$\hat{P}(a_i|c_j) = \frac{1}{e^2} \frac{\sum_{k=1}^{e^2} \hat{\boldsymbol{\Psi}}_{(D-A)_k}(j,i)}{\sum_{k=1}^{e^2} \sum_{i=1}^{m} \hat{\boldsymbol{\Psi}}_{(D-A)_k}(j,i)}, \quad (6)$$

where $\hat{\boldsymbol{\Psi}}_{(D-D)_k}$ represents posterior edge beliefs of kth $D-D$ edge and $\hat{\boldsymbol{\Psi}}_{(D-A)_k}$ represents posterior edge beliefs of kth $D-A$ edge. Using Eqn 5 and 6, we obtain the posterior conditional distribution $\hat{P}(\mathcal{C}^D|c)$ and $\hat{P}(\mathcal{C}^A|c)$ for each instance of $\mathcal{Q}$, which we call the posterior relational information.

We rely on the idea that an instance is most likely wrongly labeled by the annotator if the posterior relational information for the assigned class is not consistent with the prior relational information for that class. We use Kullback-Leibler divergence on the prior and posterior relational information and use that to assign a dissimilarity score $L = \{l_1, l_2, \ldots, l_k\}$ on each element of $\mathcal{Q}$. This dissimilarity score gives a measure of how dissimilar the prior and posterior relational information is. If i^{th} element q_i is labeled with k^{th} class c_k by the annotator, we assign the dissimilarity score to the ith instance by,

$$l_i = \frac{1}{n} \sum_{j=1}^{n} \max\Big(D_{KL}(\hat{P}(\mathcal{C}_i^D|c_k)||P(\mathcal{C}^D|c_k))$$
$$- D_{KL}(\hat{P}(\mathcal{C}_i^D|c_j)||P(\mathcal{C}^D|c_j)), 0\Big)$$
$$+ \frac{1}{m} \sum_{j=1}^{m} \max\Big(D_{KL}(\hat{P}(\mathcal{C}_i^A|c_k)||P(\mathcal{C}^A|c_k))$$
$$- D_{KL}(\hat{P}(\mathcal{C}_i^A|c_j)||P(\mathcal{C}^A|c_j)), 0\Big) \quad (7)$$

Here, $\hat{P}(\mathcal{C}_i^D|c)$ and $\hat{P}(\mathcal{C}_i^A|c)$ represent the posterior relational information of the ith instance q_i.

C. Model Update

For i^{th} labeled instance, the estimated weight from the dissimilarity score is,

$$\gamma_i = 1 - \frac{l_i}{max(l_1, l_2, \ldots, l_k)} \quad (8)$$

We consider a threshold β for detecting wrong labels. Instances for which $\gamma > \beta$, are considered correctly labeled. The classification model $\mathcal{M}^{t_0}$ and relationship model $\mathcal{R}^{t_0}$ is updated with these labels, which result in new classification model $\mathcal{M}^{t_1}$ and new relationship model $\mathcal{R}^{t_1}$.

IV. EXPERIMENTS

To evaluate the effectiveness of our proposed method, we conduct experimental analysis considering noisy labels in three different application domains: scene classification, activity classification, and document classification. These domains are selected as the data representative of the domains can share relationships among them, which is required to form the relationship model.

A. Dataset

MIT-67 Indoor [45] dataset is used for scene classification application domain. The dataset contains images of 67 indoor scene categories. We use the proposed split of trainset and testset by [45] where the trainset contains 80 images per class and the testset contains 20 images per class. For activity classification, we use VIRAT [46] dataset. It consists of 329 sequences of 11 activity classes totaling 1422 activities. For document classification, we use the CORA [47] dataset. It consists of 2708 scientific publications divided into 7 classes. These publications are linked by citations.

B. Features and Graphical Representation

Scene Classification. Scene features of MIT-67 Indoor dataset images are extracted using ResNet-50 [48] pre-trained on the Places365 [49] dataset. An off-the-shelf object detector is used to detect objects that are present in the image. We use the Matterport Mask R-CNN implementation [50], which is built on Feature Pyramid Network (FPN) [51] and a ResNet101 [48] backbone. We use a model trained on MS COCO dataset [52] to detect objects. Each image in the dataset is graphically represented by a single scene node and multiple object nodes corresponding to the objects detected in that image. Scene node potentials are obtained using the current scene classification model, which we learn incrementally with each incoming batch. Object node potentials are obtained using an off-the-shelf detector. To detect mislabeled scene nodes, we use the scene-object (S-O) relationships. All the object nodes are connected to the scene node in the graphical representation of an image forming a tree structure. We use the co-occurrence frequencies of scene classes and object classes to build the relationship model and assign edge potentials of the graph.

Activity Classification. C3D [53] model trained on sports-1M [54] dataset is used to extract features from activity segments. We extract the C3D features for every 16 frames, with a temporal stride of eight frames, and apply max-pooling to obtain a feature vector $\boldsymbol{X}_j \in \mathbb{R}^{4096}$ for each segment. Each sequence of the VIRAT dataset is represented by an undirected graph. We consider two sets of nodes: activity and object/person and two sets of edges: activity-activity and activity-object/person. We consider activities within a certain

TABLE I: Comparison of the performance of CNLD with other approaches for the noisy label detection task in symmetric label noise scenario. We compare the performance for the set of noise ratio $\Omega \in \{0.10, 0.20, 0.30, 0.40, 0.50\}$. The performance is evaluated in terms of Type-I error (ER1), Type-II error (ER2), and Noise Elimination Precision (NEP) for three datasets.

Dataset	Method	$\Omega = 0.10$			$\Omega = 0.20$			$\Omega = 0.30$			$\Omega = 0.40$			$\Omega = 0.50$		
		ER1	ER2	NEP	ER1	ER2	NEP	ER1	ER2	NEP	ER1	ER2	NEP	ER1	ER2	NEP
Scene	Majority [43]	0.10	0.88	0.12	0.19	0.76	0.24	0.29	0.68	0.32	0.37	0.57	0.43	0.47	0.45	0.55
	Consensus [43]	0.10	0.87	0.13	0.19	0.76	0.24	0.27	0.64	0.36	0.35	0.53	0.47	0.43	0.41	0.59
	Probabilistic [44]	0.09	0.81	0.19	0.19	0.76	0.24	0.26	0.61	0.39	0.34	0.52	0.48	0.43	0.43	0.58
	CNLD	**0.08**	**0.72**	**0.28**	**0.12**	**0.49**	**0.51**	**0.19**	**0.44**	**0.55**	**0.25**	**0.38**	**0.62**	**0.30**	**0.31**	**0.71**
Activity	Majority [43]	0.09	0.80	0.20	0.16	0.63	0.37	0.22	0.51	0.49	0.26	0.40	0.60	0.31	0.31	0.69
	Consensus [43]	0.08	0.71	0.19	0.14	0.54	0.46	0.18	0.41	0.59	0.20	0.31	0.69	0.24	0.23	0.77
	Probabilistic [44]	0.05	0.49	0.51	0.10	0.38	0.62	0.12	0.29	0.71	0.17	0.25	0.75	0.19	0.19	0.81
	CNLD	**0.05**	**0.45**	**0.56**	**0.09**	**0.35**	**0.65**	**0.12**	**0.29**	**0.71**	**0.16**	**0.24**	**0.76**	**0.17**	**0.18**	**0.83**
Document	Majority [43]	0.09	0.79	0.21	0.16	0.64	0.36	0.22	0.50	0.50	0.27	0.40	0.60	0.30	0.30	0.70
	Consensus [43]	0.08	0.79	0.21	0.14	0.57	0.43	0.18	0.42	0.58	0.22	0.40	0.60	0.25	0.24	0.76
	Probabilistic [44]	0.06	0.54	0.46	0.11	0.41	0.58	0.15	0.36	0.65	0.19	0.28	0.72	0.22	0.23	0.78
	CNLD	**0.05**	**0.45**	**0.55**	**0.09**	**0.34**	**0.65**	**0.11**	**0.28**	**0.73**	**0.15**	**0.22**	**0.78**	**0.18**	**0.19**	**0.82**

spatio-temporal distance to be related to each other. The co-occurrence frequencies of the activity-activity or activity-object/person within a certain spatio-temporal region are used to build the relationship model and assign the edge potentials. Activity node potentials are obtained using the current activity classifier. Object/person node potentials are obtained using the same binning approach used in [55].

Document Classification. In the CORA dataset, every publication instance is represented using a dictionary of 1433 unique words. The feature vector $X_j \in \{0,1\}^{1433}$ indicates the absence or presence of these words. We use the citation link information to build the graphs. Every publication instance is considered as a node and edges are formed when an instance is linked with another one via a citation. Node potentials are obtained using the current document classifier. Edge potentials are represented by a matrix consisting of the number of times a document of a certain class is linked to a document of another class. We also use this link information to build the relationship model.

C. Experimental Setups

We conduct experiments to analyze both the performance of our proposed Context-aware Noisy Label Detection approach for detecting wrong labels and the proposed active learning framework for robust classification in all three application domains. We use two different experimental setups to analyze these two different tasks. Multinomial Logistic Regression (MLR) is used as the classifier for all three applications. Note that the steps of our algorithm are independent of the particular choice of a classifier. Publicly available UGM Toolbox [56] is used to infer on the node and edge beliefs. For all classification tasks, we divide the training set into multiple batches and these batches are made available sequentially. All the performance evaluations reported here are an average of multiple rounds of experiments. For scene classification, we use the proposed split by [45] and in each round of an experiment, we form batches of unlabelled data by randomly shuffling instances. We also change the set of instances that are assigned with wrong labels randomly in each round of an experiment. For activity classification, 176 sequences (761 activities) are used for training and other 153 sequences (661 activities) are used

for testing. In each round of an experiment, we assign wrong labels to a randomly selected set of instances. For document classification, we use 10-fold cross-validation and different sets of instances are selected and assigned wrong labels in each round of an experiment.

D. Noisy Label Detection Performance Analysis

In our proposed approach, we utilize context information to detect noisy labels. The context information is encoded using a graphical representation. For noisy label detection, we divide the training set in multiple batches and use the instances from a single batch to train an initial classification model $\mathcal{M}$ and an initial relationship model $\mathcal{R}$. We consider the labels of these instances are correct. In the testset, a fraction of the instances is assigned with wrong labels, where the wrong labels are generated synthetically. To detect the wrong labels in the testset, we represent the instances from the testset graphically as described in section IV-B. We utilize the trained models and the graphical representations to calculate the dissimilarity scores, where the dissimilarity score is an indicator of how likely a label is wrong. The detection of a noisy label can be considered as a binary classification problem and the normalized dissimilarity score from Eqn. 7 can be treated as the confidence score for this binary classification task. To analyze the performance of this proposed approach, we conduct the following experiments:

- We analyze the performance of CNLD for detecting wrong labels when symmetric label noise is considered.
- We analyze the performance of CNLD for detecting wrong labels when asymmetric label noise is considered.

To evaluate the performance of noisy label detection, we use Type-I error (ER1), Type-II error (ER2), and Noise Elimination Precision (NEP) as described in [9]. Type-I error represents the percentage of correctly labeled instances that are erroneously removed. Type-II error represents the percentage of mislabeled instances which are not removed. Noise Elimination Precision measures the percentage of removed samples that are actually mislabelled. The corresponding measures are:

$$\text{ER1} = \frac{\text{\# of correctly labelled instances which are removed}}{\text{\# of correctly labelled instances}}$$

$$ER2 = \frac{\text{\# of mislabelled instances which are not removed}}{\text{\# of mislabelled instances}}$$

$$NEP = \frac{\text{\# of mislabeled instances which are removed}}{\text{\# of removed instances}}$$

Performance of CNLD for Symmetric Label Noise. Table I illustrates the performance of CNLD for noisy label detection task in symmetric noise scenario. In this experimental setup, the noisy labels are generated synthetically using the Noisy Completely at Random (NCAR) statistical model. We consider symmetric noise where Ω fraction of each class of the testing set samples are assigned with wrong labels and consider $\Omega \in \{0.10, 0.20, 0.30, 0.40, 0.50\}$. We compare our proposed approach with majority voting [43], consensus voting [43], and probabilistic approach [44]. Majority voting and consensus voting approaches use multiple classifiers to detect noisy labels. A label is detected as wrong if predictions of the majority of the classifiers disagree with the assigned label in the majority voting approach. Similarly, a label is detected as wrong if predictions of all of the classifiers disagree with the assigned label in the consensus voting approach. We use logistic regression, SVM, and kNN as classifiers for both of these approaches. All the classification models are learned using the same initial batch of correctly labeled data. In the probabilistic approach, the wrong labels are detected using the mismatch of assigned labels with classifier prediction and entropy of the class prediction.

In our experimental setup, we detect Ω fraction of the testing set as wrong labels for all approaches and compute Type-I error (ER1), Type-II error (ER2), and Noise Elimination Precision (NEP) scores. Here, low ER1 and ER2 scores and high NEP scores indicate better performance. For all three application domains and different noise rate Ω, there is a significant improvement in performance for our proposed approach. We observe a maximum of 26%, 5%, and 9% absolute improvement in NEP scores in scene, activity, and document dataset respectively. In the scene dataset, NEP scores for majority voting and consensus voting approach are close to the value of noise rate (Ω), indicating the inefficacy of the two approaches for noisy label detection. Compared to the activity and document dataset, we observe that the difference between noisy label detection performance of the baseline approaches and CNLD in the scene dataset is more significant. This is because, the three compared noisy label detection approaches solely rely on the performance of the learned classifiers, while CNLD relies on both the classifier and the context information. In the scene dataset, the learned classifier has less accuracy compared to the activity and document dataset, resulting in poor detection performance for the three compared approaches while CNLD retains a good detection performance by utilizing contextual information.

Performance of CNLD for Asymmetric Label Noise. Table II illustrates the performance of CNLD for noisy label detection task in an asymmetric noise scenario. We use the Noisy at Random (NAR) model to generate asymmetric label noise synthetically. The transition probabilities of the label transition matrix Λ are calculated using k-means clustering. For a dataset with n classes, we initialize n cluster centers,

TABLE II: Comparison of the performance of CNLD with other approaches for the noisy label detection task in asymmetric label noise scenario. We observe an improvement of performance for our proposed CNLD in all three datasets.

Dataset	Method	ER1	ER2	NEP
Scene	Majority [43]	0.20	0.77	0.23
	Consensus [43]	0.20	0.75	0.25
	Probabilistic [44]	0.19	0.73	0.27
	CNLD	**0.17**	**0.63**	**0.37**
Activity	Majority [43]	0.27	0.41	0.59
	Consensus [43]	0.21	0.32	0.68
	Probabilistic [44]	0.26	0.31	0.69
	CNLD	**0.16**	**0.24**	**0.76**
Document	Majority [43]	0.24	0.41	0.59
	Consensus [43]	0.21	0.36	0.64
	Probabilistic [44]	0.18	0.30	0.70
	CNLD	**0.10**	**0.18**	**0.82**

where each cluster center is the calculated average of features from all the samples of a class. Then we use the assignment step and update step to update the cluster centers until convergence. We consider a cluster to be representative of a particular class if it contains most samples from that class. If a cluster represents class y, then we calculate the transition probabilities $P(\tilde{Y} = \tilde{y}|Y = y)$ based on the number of samples that cluster contains from class $\tilde{y}$. We obtain 10%, 7%, and 12% absolute improvement in NEP scores compared to the best performing baseline approach in scene, activity, and document dataset respectively. We observe that for asymmetric label noise, the difference in performance between baseline approaches and our proposed approach is large compared to the symmetric noise scenario in activity classification and document classification. This indicates that compared to other approaches, CNLD is able to retain the detection performance in the asymmetric case.

E. Classification Robustness Analysis

To analyze the performance of our proposed framework for robust classification in an active learning setup, the entire training set is divided into multiple batches. These batches of data become available sequentially. We assume that all the instances of the initial batch are correctly labeled and data from other batches are unlabeled. The initial batch is used to learn the initial classification model $\mathcal{M}$ and the initial relationship model $\mathcal{R}$. When an unlabelled batch is available, for each unlabeled batch of data, an informative subset of instances is selected and queried for manual labeling. After obtaining the labels from an annotator, a fraction of the annotated instances is randomly selected and assigned wrong labels. To detect the incorrect labels, we represent the data graphically as described in section IV-B and calculate the dissimilarity scores. Following III-C, we discard the wrong labels and incrementally update the models with detected correct labels. We evaluate the classification performance on the same testing set whenever the model is updated. We divide the scene and document training set in 10 batches and the activity training set in 9 batches. To analyze the performance of our proposed framework, we conduct the following experiments:
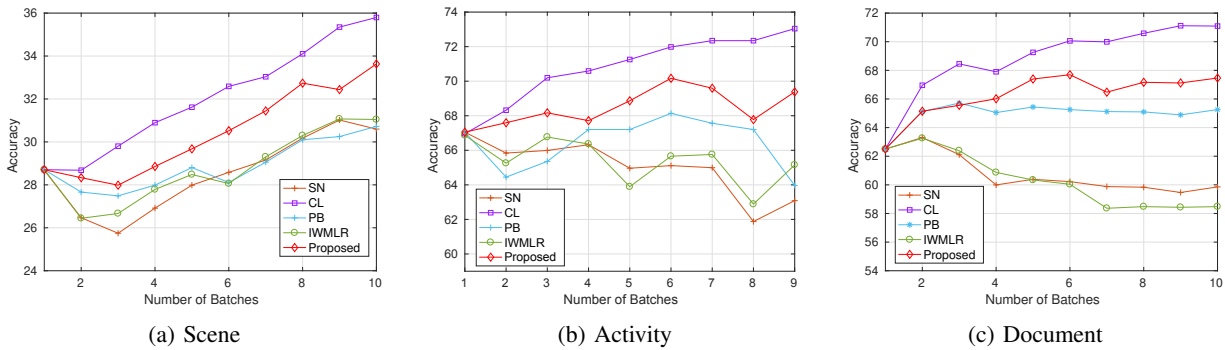
(a) Scene   (b) Activity   (c) Document

Fig. 3: Comparison of the classification performance of our proposed framework for active learning setup for three different applications: scene classification, activity classification, and document classification. We compare our proposed framework with two baseline approaches (SN, CL) and two noise-robust approaches (PB, IWMLR) for $\Omega = 0.40$ and observe superior performance in all three applications.
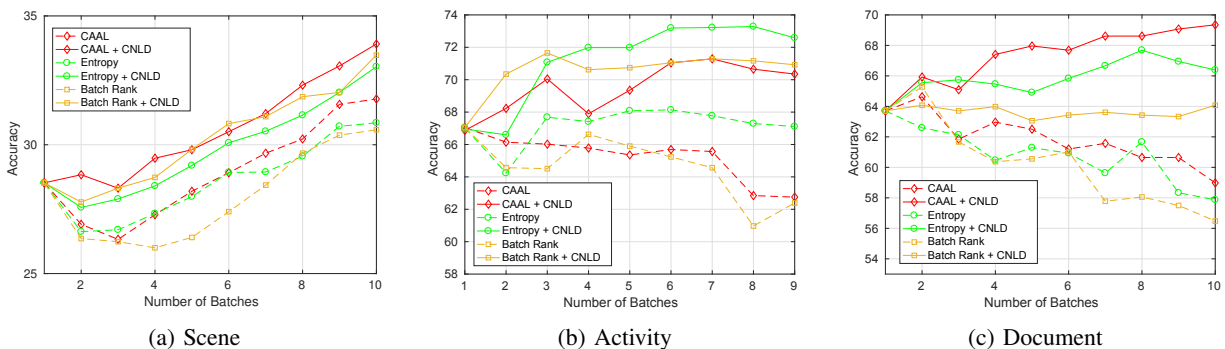

(a) Scene   (b) Activity   (c) Document

Fig. 4: Analysis of the robustness of three different active learning approaches when combined with the proposed Context-aware Noisy Label Detection (CNLD) framework for $\Omega = 0.40$. The figures illustrate that in all three active learning strategies, accuracy is higher when the active learning strategies are combined with CNLD.

- We compare the performance of our proposed active learning framework with two baseline approaches and two other label noise-robust approaches for the classification task.
- We verify the robustness of different active learning strategies combined with our proposed CNLD approach for the classification task.
- We consider a similar incremental setup with pseudo labeling and analyze the impact of CNLD for robust learning.

Classification Performance Comparison. Figure 3 illustrates the classification performance of two baseline approaches and two other label noise-robust approaches compared to our proposed approach. In this experimental setup, we consider symmetric noise with $\Omega = 0.40$ and synthetically assign wrong labels on the queried samples using the NCAR model. We compare our proposed method with the following learning approaches:

⋄ **SN**: In this approach, when the unlabelled batch of data becomes available, the classification model is updated with manually queried labels from the batch. Here, the queried labels are noisy.

⋄ **PB**: This is also an incremental learning approach. We use the probabilistic approach as discussed in IV-D to detect wrong labels of the queried samples from an unlabelled batch of data and discard them. Then the classification

model is updated using the rest of the labels.

⋄ **CL**: In this setup, we utilize the ground truth information of which labels are wrong. The classification model is updated by discarding the wrong labels. This approach represents the upper bound of the classification performance.

⋄ **IWMLR**: State-of-the-art non-deep learning noise resilient method namely Importance Weighted Multinomial Logistic Regression (IWMLR) [15]. Note that IWMLR is proposed for multi-class learning when all the data are available. We adapt it to active learning for proper comparison. When updating a classification model, for each of the queried labels, this method assigns a weight on the sample based on the likelihood of the label is wrong. It enables the learning on noisy data to more closely reflect the results on learning noise-free data.

Note that we do not compare with the Multi-class Multi-annotator Robust Gaussian Process (RGP) [14]. The reasons for this are as follows. i) In the addressed dataset, Gaussian Process Classification (GPC) performs poorly compared to the parametric Multinomial Logistic Regression (MLR). In all three applications, MLR trained with the labels from the initial batch of data performs with higher accuracy compared to GPC. ii) RGP is used in a multi-annotator setting and shown to have a time complexity of $\mathcal{O}(n^3)$ [57], which makes the

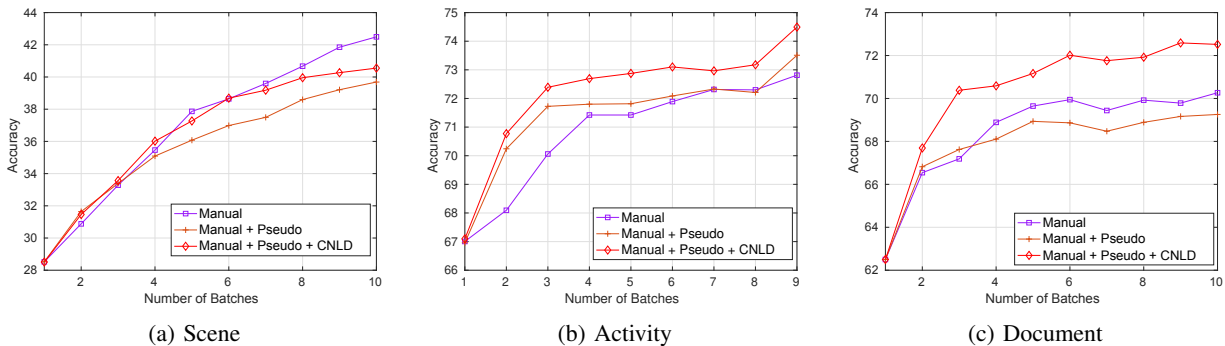(a) Scene   (b) Activity   (c) Document

Fig. 5: Comparison of classification performance with pseudo labels for three different applications: scene, activity, and document classification. We analyse how pseudo labeling combined with CNLD can help improve performance.

approach not suitable for the addressed datasets. In PB and CL approach, we discard the same number of labels as our proposed approach to make a fair comparison.

For all three classification tasks, SN illustrates the negative consequence of label noise on the learning process. CL illustrates the upper bound of the classification performance by discarding the wrong labels. We do not observe much improvement for IWMLR compared to SN for these datasets. Compared to the scene classification task, PB performs better than SN in both activity and document classification tasks. This is apparent as the noisy label detection performance of the probabilistic approach is better for activity and document classification. However, the plots in Fig. 3 illustrate that during the active learning process, our proposed method retains better accuracy than other approaches for each stage of updating the model with a new set of noisy labels. We provide the performance comparison of our proposed approach with SN for $\Omega \in \{0.20, 0.30, 0.50\}$ in the supplementary material.

Robustness of Different Active Learning Strategies. To demonstrate that the proposed framework is independent of the selection of an active learning strategy, we analyze the robustness of different active learning techniques when combined with CNLD. Figure 4 illustrates the robustness of our proposed approach for different active learning strategies. We select three commonly used active learning methods: Entropy [3], Batch Rank [58], and CAAL [28]. For each of the active learning approaches, we compare the performance of the classifier when updated with noisy labels vs. when updated with detected correct labels by CNLD. In Figure 4, the dotted lines refer to the classification performance of different active learning strategies where the classifier is updated with noisy labels. The solid lines in Figure 4 represent the classification performance when the active learning approach is combined with CNLD. For all three applications and for each of the active learning approaches, frameworks combined with CNLD are more robust and result in higher accuracy compared with their vanilla counterpart.

Pseudo Labeling We consider another incremental setup where we update the models with manually queried labels and pseudo labels. In this setup, the initial models are learned using the initial set of data. When a new batch of data becomes available, we select an informative set of samples and query

TABLE III: In this table, we report the improvement in performance for our proposed approach over SN for different selection of β. We show the results for $\Omega \in \{0.10, 0.20, 0.30, 0.40, 0.50\}$ and for different β selection from the set $\{0.80, 0.85, 0.90\}$.

	Noise	Accuracy improvement over SN		
		$\beta = 0.80$	$\beta = 0.85$	$\beta = 0.90$
Scene	$\Omega = 0.10$	0.24 ± 0.47	0.03 ± 0.44	-0.29 ± 0.83
	$\Omega = 0.20$	0.36 ± 0.39	0.22 ± 0.71	0.06 ± 0.71
	$\Omega = 0.30$	1.02 ± 0.70	1.00 ± 0.61	1.51 ± 0.80
	$\Omega = 0.40$	1.93 ± 0.87	1.78 ± 0.90	1.65 ± 0.70
	$\Omega = 0.50$	1.99 ± 0.90	2.46 ± 1.13	3.81 ± 1.70
Activity	$\Omega = 0.10$	0.54 ± 0.53	0.62 ± 0.59	1.23 ± 0.72
	$\Omega = 0.20$	-0.05 ± 0.79	0.34 ± 0.60	0.26 ± 0.59
	$\Omega = 0.30$	1.21 ± 0.84	1.44 ± 1.03	2.43 ± 1.76
	$\Omega = 0.40$	1.68 ± 1.57	1.57 ± 1.02	3.45 ± 2.21
	$\Omega = 0.50$	1.50 ± 1.54	2.40 ± 1.60	4.25 ± 3.03
Document	$\Omega = 0.10$	1.09 ± 0.94	0.67 ± 0.82	1.43 ± 1.25
	$\Omega = 0.20$	1.13 ± 0.72	1.57 ± 0.85	1.76 ± 0.98
	$\Omega = 0.30$	3.18 ± 1.78	2.96 ± 1.64	4.06 ± 2.14
	$\Omega = 0.40$	3.04 ± 1.49	5.06 ± 2.71	5.55 ± 2.79
	$\Omega = 0.50$	3.72 ± 1.84	4.82 ± 2.32	6.51 ± 3.44

the labels. We consider these queried labels to be correct. We also utilize the unlabelled data and update the classification models with predicted labels of the unlabelled data, which is called the pseudo labels. The generation of pseudo labels is classifier dependent and can contain a lot of noise. Here, the generation of noise is feature dependent and more closely reflects the real noise scenario. So while updating models with pseudo labels, we use CNLD to detect the wrong labels and filter them. In this experimental setup, we use three learning approaches and compare their performance:

- **Manual**: In this approach, when an unlabelled batch of data is available, we update the models with only manually queried correct labels.
- **Manual + Pseudo**: In this approach, when an unlabelled batch of data is available, we update the models with manually queried correct labels and generated pseudo labels from the rest of the unlabelled data of that batch. Here, the pseudo labels are generated using the current classification model.
- **Manual + Pseudo + CNLD**: Similar to the setup of Manual + Pseudo. Additionally, we consider utilizing CNLD to

identify pseudo labels that are wrong. Then the model is updated with manually queried correct labels and pseudo labels that are detected as correct.

Figure 5 illustrates the performance comparison of the above-mentioned learning approaches. In the scene classification task, the model updated with manually correct labels (Manual) performs best. This is because the accuracy of the learned classifier is low and generates a lot of wrong labels, which eventually degrade the classification performance if used for updating the model. However, compared to using pseudo labels directly (Manual + Pseudo), there is an improvement in performance if we filter wrong pseudo labels using CNLD (Manual + Pseudo + CNLD). In activity and document classification task, Manual performs better than Manual + Pseudo. However, in both classification tasks, performance of the filtered pseudo labeling approach (Manual + Pseudo + CNLD) is superior to Manual and Manual + Pseudo.

β Parameter Sensitivity Analysis. The selection of β parameter is a trade-off between precision and recall for noisy label detection. A process with high precision may not be able to detect a lot of incorrect labels. On the other hand, a process with high recall will detect a lot of correct labels as wrong. In both cases, the performance of a classification model will degrade. As a result, our approach requires to select β in a way to balance between these two conditions. In Table III, we analyze the performance of the classification model for a wide range of selection of β. For noise rate $\Omega \in \{0.10, 0.20, 0.30, 0.40, 0.50\}$ and $\beta \in \{0.80, 0.85, 0.90\}$, we report the absolute classification accuracy improvement of our proposed approach over the baseline SN approach. We compute the average of the difference of accuracy of our proposed approach and SN approach in each incremental update and also report the standard deviations. We observe that for a wide range of selection of β, our approach gains a performance improvement over learning with noisy labels. Improvement of performance is not significant for a low noise rate ($\Omega \in \{0.10, 0.20\}$). It is expected because a small number of noisy labels do not degrade the classification performance much. For $\Omega \in \{0.30, 0.40, 0.50\}$, there is $1\% - 3.81\%$ absolute accuracy improvement in scene classification, $1.21\% - 4.25\%$ absolute accuracy improvement in activity classification, and $3.18\% - 6.51\%$ absolute accuracy improvement in document classification for different β selection.

F. Qualitative Results

We provide two qualitative examples from the MIT-67 Indoor dataset for the noisy label detection task in Figure 6. The graphical representation is constructed using the objects that are detected by an off-the-shelf object detector. Then we utilize the graphical representation and the prior relational information to compute the dissimilarity scores. A high dissimilarity score indicates the label is likely to be wrong, while a low dissimilarity score indicates the label is correct. For example, when the top image in Figure 6 is labeled as 'kitchen' scene, the contextual relation formed with object labels and a scene label will be consistent with previously learned contextual relation. As a result, the dissimilarity score is low for 'kitchen'
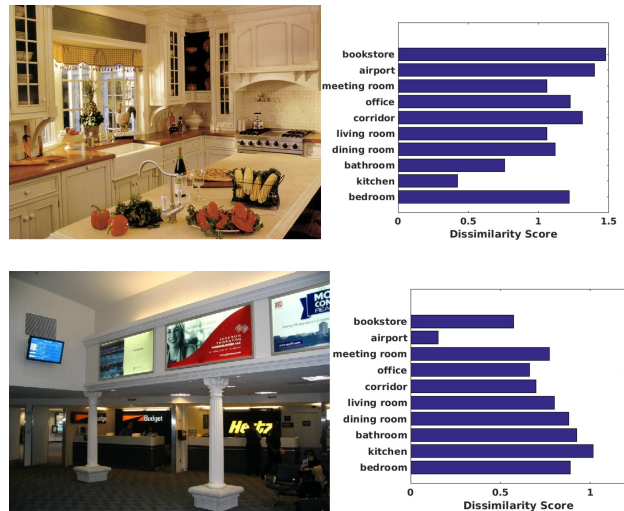

Fig. 6: Example illustration of the performance of noisy label detection. The dissimilarity score is minimum when the first image is labeled with the correct scene class 'kitchen'. Similarly, the dissimilarity score for the second image is minimum when the image is assigned with the 'airport' scene class.

class. If the image is assigned with any other label except for 'kitchen', there will be a contextual inconsistency and the dissimilarity score will be high. Similarly, for 'airport' indoor image, for 'airport' label the dissimilarity score is low while for other assigned labels, the dissimilarity score is high.

V. CONCLUSION

In this paper, we formalize a general active learning framework that utilizes a noisy label filtering based learning approach to reduce the adverse impact of label noise. In this regard, we propose a novel context-aware noisy label detection strategy. For various applications, we show how we can represent the inter-relationship among the data and using that representation, infer the likelihood of a label being noisy. The proposed noisy label robust framework is independent of a particular choice of feature, classifier, and active selection strategy. We experimentally validate the robustness of the active learning approach in the presence of label noise.

REFERENCES

[1] X. Li and Y. Guo, "Multi-level adaptive active learning for scene classification," in *European Conference on Computer Vision*. Springer, 2014, pp. 234–249.
[2] B. Settles, "Active learning," *Morgan & Claypool*, 2012.
[3] X. Li and Y. Guo, "Adaptive active learning for image classification," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2013, pp. 859–866.
[4] N. V. Cuong, W. S. Lee, N. Ye, K. M. A. Chai, and H. L. Chieu, "Active learning for probabilistic hypotheses using the maximum gibbs error criterion," in *Advances in Neural Information Processing Systems(NIPS)*, 2013, pp. 1457–1465.

[5] C. Käding, A. Freytag, E. Rodner, A. Perino, and J. Denzler, "Large-scale active learning with approximations of expected model output changes," in *German Conference on Pattern Recognition(GCPR)*. Springer, 2016, pp. 179–191.

[6] J. Sourati, M. Akcakaya, D. Erdogmus, T. K. Leen, and J. G. Dy, "A probabilistic active learning algorithm based on fisher information ratio," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 8, pp. 2023–2029, Aug 2018.

[7] B. Zhang, L. Li, S. Yang, S. Wang, Z.-J. Zha, and Q. Huang, "State-relabeling adversarial active learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 8756–8765.

[8] D. Gudovskiy, A. Hodgkinson, T. Yamaguchi, and S. Tsukizawa, "Deep active learning for biased datasets via fisher kernel self-supervision," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9041–9049.

[9] B. Frénay and M. Verleysen, "Classification in the presence of label noise: a survey," *IEEE transactions on neural networks and learning systems*, vol. 25, no. 5, pp. 845–869, 2014.

[10] X. Zhu and X. Wu, "Class noise vs. attribute noise: A quantitative study," *Artificial Intelligence Review*, vol. 22, no. 3, pp. 177–210, Nov 2004. [Online]. Available: https://doi.org/10.1007/s10462-004-0751-8

[11] G. Hua, C. Long, M. Yang, and Y. Gao, "Collaborative active learning of a kernel machine ensemble for recognition," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 1209–1216.

[12] C. Long, G. Hua, and A. Kapoor, "Active visual recognition with expertise estimation in crowdsourcing," in *2013 IEEE International Conference on Computer Vision*, Dec 2013, pp. 3000–3007.

[13] C. Long, G. Hua, and A. Kapoor, "A joint gaussian process model for active visual recognition with expertise estimation in crowdsourcing," *International journal of computer vision*, vol. 116, no. 2, pp. 136–160, 2016.

[14] C. Long and G. Hua, "Multi-class multi-annotator active learning with robust gaussian process for visual recognition," in *International Conference on Computer Vision (ICCV)*, 2015.

[15] R. Wang, T. Liu, and D. Tao, "Multiclass learning with partially corrupted labels," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 6, pp. 2568–2580, June 2018.

[16] B. Yao and L. Fei-Fei, "Modeling mutual context of object and human pose in human-object interaction activities," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2010, pp. 17–24.

[17] Z. Wang, Q. Shi, and C. Shen, "Bilinear programming for human activity recognition with unknown mrf graphs," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[18] M. J. Choi, J. J. Lim, A. Torralba, and A. S. Willsky, "Exploiting hierarchical context on a large database of object categories," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2010, pp. 129–136.

[19] C. Galleguillos, A. Rabinovich, and S. Belongie, "Object categorization using co-occurrence, location and appearance," in *IEEE Conference on Computer Vision and Pattern Recognition(CVPR)*. IEEE, 2008, pp. 1–8.

[20] M. Bilgic and L. Getoor, "Link-based active learning," in *NIPS Workshop on Analyzing Networks and Learning with Graphs*, 2009.

[21] B. Settles and M. Craven, "An analysis of active learning strategies for sequence labeling tasks," in *EMNLP*, 2008.

[22] P. M. Long and R. A. Servedio, "Random classification noise defeats all convex potential boosters," *Machine Learning*, vol. 78, no. 3, pp. 287–304, Mar 2010. [Online]. Available: https://doi.org/10.1007/s10994-009-5165-z

[23] X. Li, R. Guo, and J. Cheng, "Incorporating incremental and active learning for scene classification," in *Machine Learning and Applications (ICMLA), 2012 11th International Conference on*, vol. 1. IEEE, 2012, pp. 256–261.

[24] S. Vijayanarasimhan and K. Grauman, "Large-scale live active learning: Training object detectors with crawled data and crowds," *International Journal of Computer Vision*, vol. 108, no. 1-2, pp. 97–114, 2014.

[25] A. Vezhnevets, J. M. Buhmann, and V. Ferrari, "Active learning for semantic segmentation with expected change," in *IEEE Conference on Computer Vision and Pattern Recognition(CVPR)*. IEEE, 2012, pp. 3162–3169.

[26] W. Cai, Y. Zhang, and J. Zhou, "Maximizing expected model change for active learning in regression," in *International Conference on Data Mining(ICDM)*. IEEE, 2013, pp. 51–60.

[27] O. Mac Aodha, N. D. Campbell, J. Kautz, and G. J. Brostow, "Hierarchical subquery evaluation for active learning on a graph," in *Computer Vision and Pattern Recognition(CVPR)*, IEEE. IEEE, 2014, pp. 564–571.

[28] M. Hasan, S. Paul, A. I. Mourikis, and A. K. Roy-Chowdhury, "Context-aware query selection for active learning in event recognition," *IEEE transactions on pattern analysis and machine intelligence*, 2018.

[29] O. Sener and S. Savarese, "Active learning for convolutional neural networks: A core-set approach," *arXiv preprint arXiv:1708.00489*, 2017.

[30] Y. Gal, R. Islam, and Z. Ghahramani, "Deep bayesian active learning with image data," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2017, pp. 1183–1192.

[31] D. Yoo and I. S. Kweon, "Learning loss for active learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 93–102.

[32] S. Sinha, S. Ebrahimi, and T. Darrell, "Variational adversarial active learning," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 5972–5981.

[33] M. Kääriäinen, "Active learning in the non-realizable case," in *International Conference on Algorithmic Learning Theory*. Springer, 2006, pp. 63–77.

[34] M. Naghshvar, T. Javidi, and K. Chaudhuri, "Bayesian active learning with non-persistent noise," *IEEE Transactions on Information Theory*, vol. 61, no. 7, pp. 4080–4098, July 2015.

[35] R. M. Castro and R. D. Nowak, "Minimax bounds for active learning," *IEEE Transactions on Information Theory*, vol. 54, no. 5, pp. 2339–2353, 2008.

[36] S. Dasgupta, D. J. Hsu, and C. Monteleoni, "A general agnostic active learning algorithm," in *Advances in neural information processing systems*, 2008, pp. 353–360.

[37] A. Beygelzimer, D. J. Hsu, J. Langford, and T. Zhang, "Agnostic active learning without constraints," in *Advances in Neural Information Processing Systems 23*, J. D. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. S. Zemel, and A. Culotta, Eds. Curran Associates, Inc., 2010, pp. 199–207. [Online]. Available: http://papers.nips.cc/paper/4014-agnostic-active-learning-without-constraints.pdf

[38] C. Zhang and K. Chaudhuri, "Beyond disagreement-based agnostic active learning," in *Advances in Neural Information Processing Systems*, 2014, pp. 442–450.

[39] M.-F. Balcan, A. Beygelzimer, and J. Langford, "Agnostic active learning," *Journal of Computer and System Sciences*, vol. 75, no. 1, pp. 78 – 89, 2009, learning Theory 2006. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0022000008000652

[40] M.-F. Balcan and P. Long, "Active and passive learning of linear separators under log-concave distributions," in *Conference on Learning Theory*, 2013, pp. 288–316.

[41] S. Hanneke, "Teaching dimension and the complexity of active learning," in *International Conference on Computational Learning Theory*. Springer, 2007, pp. 66–81.

[42] J. Yao, S. Fidler, and R. Urtasun, "Describing the scene as a whole: Joint object detection, scene classification and semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2012, pp. 702–709.

[43] C. E. Brodley and M. A. Friedl, "Identifying mislabeled training data," *Journal of artificial intelligence research*, vol. 11, pp. 131–167, 1999.

[44] J.-w. Sun, F.-y. Zhao, C.-j. Wang, and S.-f. Chen, "Identifying and correcting mislabeled training instances," in *Future generation communication and networking (FGCN 2007)*, vol. 1. IEEE, 2007, pp. 244–250.

[45] A. Quattoni and A. Torralba, "Recognizing indoor scenes," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[46] S. Oh, A. Hoogs, A. Perera, N. Cuntoor, C.-C. Chen, J. T. Lee, S. Mukherjee, J. Aggarwal, H. Lee, L. Davis *et al.*, "A large-scale benchmark dataset for event recognition in surveillance video," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2011, pp. 3153–3160.

[47] P. Sen, G. Namata, M. Bilgic, L. Getoor, B. Gallligher, and T. Eliassi-Rad, "Collective classification in network data," *AI magazine*, vol. 29, no. 3, p. 93, 2008.

[48] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[49] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba, "Places: A 10 million image database for scene recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 6, pp. 1452–1464, 2018.

[50] W. Abdulla, "Mask r-cnn for object detection and instance segmentation on keras and tensorflow," 2017.

[51] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2117–2125.

[52] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*. Springer, 2014, pp. 740–755.

[53] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *International Conference on Computer Vision (ICCV)*. IEEE, 2015, pp. 4489–4497.

[54] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[55] M. Hasan and A. K. Roy-Chowdhury, "Context aware active learning of activity recognition models," in *IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 4543–4551.

[56] M. Schmidt, "Ugm: Matlab code for undirected graphical models," *URL https://www.cs.ubc.ca/ schmidtm/Software/UGM.html*, 2012.

[57] D. Hernández-Lobato, J. M. Hernández-Lobato, and P. Dupont, "Robust multi-class gaussian process classification," in *Advances in neural information processing systems*, 2011, pp. 280–288.

[58] S. Chakraborty, V. Balasubramanian, Q. Sun, S. Panchanathan, and J. Ye, "Active batch selection via convex relaxations with guaranteed solution bounds," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 10, pp. 1945–1958, 2015.