Duo Learning for Classifications with Noisy Labels*

Sophie Cerf
Univ. Grenoble Alpes
France
sophie.cerf@gipsa-lab.fr

Robert Birke
ABB Research
Switzerland
robert.birke@ch.abb.com

Lydia Y. Chen
TU Delft
Netherlands
y.chen-10@tudelft.nl

Abstract

While the vast amount of user-generated data powers up machine learning based applications in our daily life, the variation of data quality may undermine the effectiveness of learning. In this paper, we demonstrate how classification algorithms can possibly be deteriorated by incorporating the data influx with noisy labels in the context of continuous learning. We design duo-learning framework that adaptively learn the dynamics of underlying data quality in junction with an on-line classification model. Our objective is to maximize the accuracy improvement rate from the initial data by actively selecting minimum amounts of "clean" data over time. The problem generalizes across multiple application scenarios of continuous learning with noisy label data that are intrinsic or adversarial in the label generating process. Our preliminary results show that effective data selection can achieve a good accuracy improvement using a fraction amount of data instances, and avoid the pitfall of divergence from continuously learning from noisy label data.

1 Introduction

The vast amount of user-generated data, files, and images greatly enhances the applicability of machine learning technologies in our daily life. Indeed, datasets openly available in the wild come in different qualities, due to unsophisticated and unreliable processes of data generation and label annotation, or malicious adversaries. For example, similar images from Google search could show different classification labels [12]. When continuously learning from such data that is subject to label noises, the arising question is its impact on the classification algorithm over time or alternatively its associated prices/values. Let us take the crowd sourcing example: how to determine the price for annotating images to be paid to users that can produce more accurate class labels than others.

The specific question considered in the paper is the following. How to build a classification model that intelligently selects the data and continuously improves the accuracy from a minimum amount of data instances that are subject to fluctuating label noises? The challenge lies in capturing the dynamics of data quality which is not directly observable but only via classification outcomes that in turn is coupled with the noise level of data labels. When encountering noisy data labels, the classification errors are thus attributed to the classification models as well as intrinsic wrong inputs. Moreover, the noise levels can vary significantly across datasets produced by different individuals or robots. Using data with noisy labels may degrade the classification results and slow down the learning speed in the long run, or worse, make it diverge.

In this paper, we develop an on-line duo learning framework that continuously learns both data label model and classifier from arriving datasets that are subject to label noises. Our objective is to learn the classifier efficiently from a minimum amount of non-noisy data instances. For every batch of new data, we solicit data instances with non-noisy labels based on the label model that predicts if the sample unit is noisy. Our preliminary results on the random forests classifier on five datasets show that continuously cleansing data can result into a better learning efficiency, i.e., the improvement of classifier accuracy per additional dataset selected, compared to classifiers without continuous data cleansing.

*This work is partly funded by the Swiss National Science Foundation NRP75 project 407540_167266.

Preprint. Work in progress.

1.1 Related work

The prior art has extensively addressed the noisy label issues with respect to different classification algorithms [5] for static datasets in an off-line setting, i.e., the classification models are learned from a single batch of data. Noisy labels are shown to degrade the effectiveness of typical classification models, e.g., k-NN [13] and SVM [1] but also deep neural network [11]. The challenges involved include missing information of label quality, and inter dependency between classification errors and label errors that can be intrinsic or malicious.

There are two types of approaches to address classification problems with noisy labels: (i) cleansing noisy labels and learning only from the predicted clean data instances, (ii) designing noise-aware classification algorithms. The first builds one or multiple filter models, e.g., SVM [9] to cleanse the data, and only the data instances that have been correctly classified by one or more filters are used to train the classification models. In contrast, the second type of approach tries to capture the noise model via frequentist [6] or Bayesian methods [10] and incorporate it in the underlying classification algorithms, e.g., adjusting the loss function. [3] applies large dual weights on different training sample units when learning a SVM classifier. In [7], the metric of local intrinsic dimensionality is used to control the training of deep neural networks.

While there exist good solutions to tackle the noisy label issues, little focus has been given to the on-line setting with data instances having fluctuating noise levels. Our study presents the initial results on how to build a classifier by selectively and continuously learn from high quality data that leads to a strong classifier.

2 Methodology

We specifically consider the following problem statement. A small set of initial data instances that have clean labels is given, and another set of testing data is provided. Each data instance has d features and belongs to class k, where $k \in \mathbf{K} = \{\mathbf{1}, \ldots, K\}$. A clean label refers to samples whose given label is properly annotated, without any alteration. The data instances are continuously collected over time, subject to a certain constraint. Specifically, we assume only up to N instances can be considered for training classifier. Furthermore, data instances are assumed to arrive at the learning system in batches over time X_i and the quality of their labels $\tilde{y}_i$ may fluctuate, e.g., percentage of noisy labels changes from batches to batches. The question here is how to continuously train a classifier from live noisy data so that the accuracy improvement from the initial set can be improved given the constraint of N data instances.

To such an end, we develop a duo learning framework as shown in Fig. 1 that continuously trains the label model, $\mathfrak{L} : \mathbb{R}^d \rightarrow q \in \mathbf{Q} = \{\mathbf{0}, \mathbf{1}\}$, and the classification model, $\mathfrak{C} : \mathbb{R}^d \rightarrow k \in \mathbf{K}$. The label model tries to learn the binary classifier for label quality, i.e., $q = 0$ denotes the clean label and $q = 1$ otherwise. Upon the arrival of a new batch of data instances at epoch i, we use $\mathfrak{L}_{i-1}$ to predict the quality for each data instance j. Here we use the subscript of i to denote the label model that is trained with data instance received up to time i. If $\hat{q}_{j,i} = 1$, meaning a noisy label, we discard such a data instance and only incorporate data instances with $\hat{q}_{j,i} = 0$ into the existing training set to retrain the classifier, $\mathfrak{C}_i$. Essentially, $\mathfrak{C}_i$ is trained on all the estimated-clean data received after the arrival of i^{th} batch. In turn, we retrain the label model for the next batch $\mathfrak{L}_i$ by incorporating those clean data, indicated by the black feedback arrow.

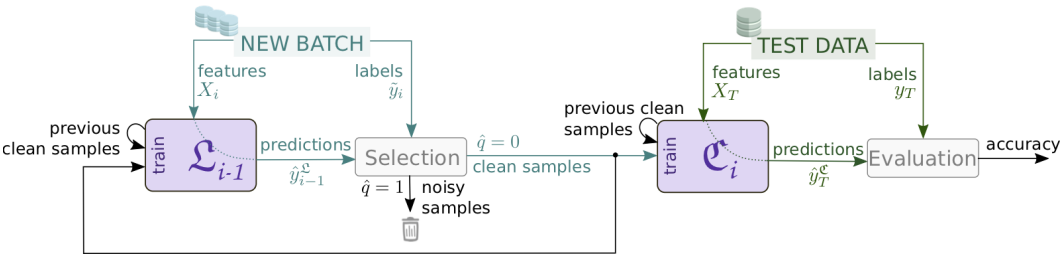


Figure 1: Duo learning framework. The flowchart is iterated at every batch.

The objectives of label model is two fold here. First of all, the label model can select the most representative instances to learn a strong classifier. The second-order objective here is to solicit data instances with clean labels, avoiding the pitfall that the classifier overfits the noise. To achieve both ends, we advocate to use supervised-learning algorithms for continuously training label model from accumulated "predicted" clean samples, which are highly correlated to a stronger classifier. Upon the arrival of data instances at batch i, we apply the previously learned label model $\mathfrak{L}_{i-1}$ to first predict their class labels $\hat{y}_i^{\mathfrak{L}}$. If predicted labels are different from their given labels, we then label such instances noisy. We then preclude them from the training set and only use the remaining subset of instances for retraining the classifier $\mathfrak{C}_i$. Eventually, the classification accuracy is measured by comparing test labels y_T to the predictions $\hat{y}_T^{\mathfrak{C}}$ based on test data X_T.

The proposed duo learning framework is generic and compatible with different learning algorithms for label and classifier models. In the evaluation section, we consider the following learning algorithms: knn, nearest centroid, SVM, Gaussian process and random forests. Considering the practical applicability on live data that might face timing constraint, we recommend to use lower order model for learning label quality than the classifier.

3 Evaluation

3.1 Experimental Setup

To evaluate the proposed duo learning, we use the following datasets: *dna*, *letter*, *pendigits*, *usps* and *mushrooms* from [2] and [4]. They vary in number of classes, features and number of samples. Their details are summarized in Table 1.

Table 1: Summary of the datasets main properties

Dataset	dna	letter	pendigits	usps	mushrooms
#-classes k	3	26	10	10	2
#-features d	180	16	16	256	24

The continual learning scenario, common for all datasets, is the following. Initially, a set of 150 clean samples is available and the testing set is clean. Then, data batches arrive continuously in sets of 50 sample units whose class labels may be altered and contains noises. We assume that there is an upper limit of $N = 1000$ samples (including the initial set) to be considered for training classifier. This number is chosen so as to ensure that the classification accuracy can converge in absence of noise for all datasets.

We add symmetric noises, i.e. following the *noise completely at random* model [5]. Time-varying dynamic noise is considered, the mean noise ratio (percentage of noisy labels in a batch) is drawn at every batch from a Gaussian distribution, with 20% of standard deviation. The mean rate is set depending on the number of classes in the dataset. We specifically consider the challenging scenarios where noisy labels affect the classification model. When the set has many classes (e.g.,*letter*, *pendigits* and *usps*), the learning is inherently more robust to noise, so the mean noise rate is set to 80%. For the datasets with fewer classes (*dna* and *mushrooms*), mean noise rate is set to 40%.

We use random forest for the classification model for multiple reasons. Random forest is applied on a variety of real applications, and is also known to be robust against label noises [5]. For fair comparison its parametrization is fixed to 100 trees, max-*depth* of 5 and *max-features* of 10. It is indeed not the optimal configuration for all datasets but it provides reasonable results on clean sets.

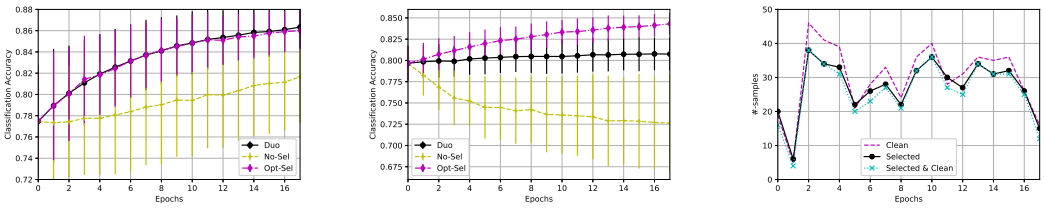
The label model is nearest centroid. We tested other learning algorithms, e.g. SVM or Gaussian process, but the simpler neighbor-based ones, e.g., nearest neighbors or nearest centroid, have shown better performance. The simple models remove not only noisy labels but also outliers. It hence increases the similarity between selected samples and thus classification robustness.

The framework is developed in Python using *scikit-learn* [8]. The main evaluation metric is the accuracy score, averaged over 100 runs. The proposed duo learning is compared against two other data selection schemes: (i) *No-Sel*, where all data instances of arriving batches are used for training classifier, (ii) *Opt-Sel*, which emulates an omniscient agent who can perfectly distinguish between clean and noisy labels.

3.2 Results

Fig. 2 (a) and (b) show the accuracy of the classification model on the *dna* and *usps* test sets for successive batches. The proposed learning framework (Duo) is compared against no selection (No-Sel) and optimal selection (Opt-Sel). These plots highlight four main results: (i) there is an advantage of continual learning compared to using only the initial set, (ii) however, learning from noisy labels over time leads to a serious accuracy degradation for the classifier, (iii) the duo learning improves the classification accuracy compared to taking all labels, (iv) the data selection of the duo learning is not far from being the optimal one.

Fig. 2 (c) presents the variations of noise over time (number of clean samples) for one run on *dna*. Overlaid is the number of data selected by the duo learning and the overlap between selection and actually clean. Results highlight the sharpness of data selection and its parsimony.



(a) Accuracy for *dna* (b) Accuracy for *usps* (c) Data selection for *dna*

Figure 2: Duo learning framework performances through time.

A full evaluation on all datasets is reported in Table 2. In addition to the average accuracy after all epochs, we also present the percentage of accuracy improvement comparing the duo to the initial set and no selection in columns 5 and 6 in Table 2. We attribute columns 5 and 6 to the impact of continual learning and intelligent selection respectively. As for columns 7 and 8, we present the percentage of accuracy difference between the duo and Opt-Sel (the smaller the better) and the fraction of samples used by duo.

Table 2: Evaluation of the duo learning framework for all datasets. Negative results are in **bold**.

Dataset	Initial accuracy	Final accuracy			Improvement due to		Distance to optimal	Samples used
		No-Sel	Duo	Opt-Sel	Continual	Selection		
dna	77.71	82.25	86.06	86.02	10.7%	4.6%	0%	55.6%
letter	43.76	41.04	42.78	52.51	**-2.2**%	4.2%	18.5%	10.5%
pendigits	81.64	77.05	80.07	84.87	**-1.9**%	3.9%	5.6%	19.6%
usps	79.81	73.21	81.03	84.46	1.5%	10.6%	4%	20.30%
mushrooms	97.33	94.22	80.74	98.94	**-17**%	**-14.3** %	18.3%	55.7%

Most results are positive, with varying amplitude depending on the datasets. In most cases, the proposed duo improves accuracy compared to blindly taking all samples, from 4 to 10% with the exception of *mushrooms* dataset. The impact of continual learning is more variable. It is highly valuable for *dna*, small but positive impact for *usps*, but detrimental for *pendigits*, *letter* and *mushrooms*. The poor performance of some sets is not linked with either their number of classes nor features. The optimal selection performs indeed better than the proposed duo, however for *dna* the quality estimation is close to optimal. Eventually, the total number of samples used ranges from half of the dataset to a tenth, which implies great potential savings in the scenario of data purchasing. The distance to optimal metric and *mushrooms* case show that there is still room for improvement.

4 Conclusion

In this paper we develop a duo learning framework that can continuously cleanse the live data instances with fluctuating noise levels and effectively learn a strong classifier from a small fraction of data. Our initial results show that even a simple label model, such as nearest centroid, can effectively select data instances and improve the classification accuracy and learning efficiency for random forests classifier. Our future work will explore a wider set of classification algorithms, including deep neural networks for image classification.

References

[1] Wenjuan An and Mangui Liang. Fuzzy support vector machine based on within-class scatter for classification problems with outliers or noises. *Neurocomput.*, 110:101–110, June 2013. ISSN 0925-2312.

[2] Chih-Chung Chang and Chih-Jen Lin. Libsvm: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.*, 2(3):27:1–27:27, May 2011. ISSN 2157-6904. doi: 10.1145/1961189.1961199. URL `http://doi.acm.org/10.1145/1961189.1961199`.

[3] Ofer Dekel and Ohad Shamir. Good learners for evil teachers. In *Proceedings of the 26th Annual International Conference on Machine Learning, ICML 2009, Montreal, Quebec, Canada, June 14-18, 2009*, pages 233–240, 2009.

[4] Ayhan Demiriz (demira@rpi. edu, Kristin P Bennett, John Shawe, and Taylor jst@cs. rhul. ac. uk. Linear programming boosting via column generation. (bennek@rpi.edu).

[5] Benoît Frénay and Michel Verleysen. Classification in the presence of label noise: A survey. *IEEE Trans. Neural Netw. Learning Syst.*, 25(5):845–869, 2014.

[6] Yunlei Li, Lodewyk F. A. Wessels, Dick de Ridder, and Marcel J. T. Reinders. Classification in the presence of class noise using a probabilistic kernel fisher method. *Pattern Recognition*, 40(12):3349–3357, 2007.

[7] Xingjun Ma, Yisen Wang, Michael E. Houle, Shuo Zhou, Sarah M. Erfani, Shu-Tao Xia, Sudanthi N. R. Wijewickrema, and James Bailey. Dimensionality-driven learning with noisy labels. In *ICML 2018*, pages 3361–3370.

[8] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[9] Nicola Segata, Enrico Blanzieri, Sarah Jane Delany, and Padraig Cunningham. Noise reduction for instance-based learning with a local maximal margin approach. *J. Intell. Inf. Syst.*, 35(2): 301–331, 2010.

[10] James D. Stamey and Richard Gerlach. Bayesian model selection for logistic regression with classified outcomes. *Statist. Model*, 7(3):255–273, 2007.

[11] Arash Vahdat. Toward robustness against label noise in training deep discriminative neural networks. In *NIPS*, pages 5601–5610, 2017.

[12] Yisen Wang, Weiyang Liu, Xingjun Ma, James Bailey, Hongyuan Zha, Le Song, and Shu-Tao Xia. Iterative learning with open-set noisy labels. In *CVPR 2018*.

[13] D. Randall Wilson and Tony R. Martinez. Reduction techniques for instance-basedlearning algorithms. *Mach. Learn.*, 38(3):257–286, March 2000.