RT2S: A Framework for Learning with Noisy Labels

Indranil Bhattacharya
bindrani@amazon.com
Amazon.com, Inc.
Seattle, WA, USA

Ze Ye
yeze@amazon.com
Amazon.com, Inc.
Seattle, WA, USA

Kaushik Pavani
sripava@amazon.com
Amazon.com, Inc.
Seattle, WA, USA

Sunny Dasgupta
sunnyd@amazon.com
Amazon.com, Inc.
Seattle, WA, USA

ABSTRACT

We introduce Robust Training with Trust Scores (RT2S), a framework to train machine learning classifiers with potentially noisy labels. RT2S calculates a trust score for each training sample, which indicates the quality of its corresponding label. These trust scores are employed as sample weights during training and optionally during threshold optimization. The trust scores are generated from two sources: (i) the model's confidence in the observed label, leveraging out-of-fold prediction scores to detect anomalous labels in the training data, and (ii) the probability of the correct label, ascertained by a Large Language Model with the ability to identify biased label noise. We evaluate RT2S by training machine learning models on 6 product classification datasets that utilize low-quality labels generated by a rule-based classification engine acting as a surrogate labeler. Our experimental findings indicate that RT2S outperforms all baselines, and achieves an average accuracy improvement of 4.38% (max 7.18%) over rule-based classifiers in particular.

ACM Reference Format:
Indranil Bhattacharya, Ze Ye, Kaushik Pavani, and Sunny Dasgupta. 2023. RT2S: A Framework for Learning with Noisy Labels. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management (CIKM '23), October 21–25, 2023, Birmingham, United Kingdom.* ACM, Birmingham, UK, 2 pages. https://doi.org/10.1145/3583780.3615996

1 INTRODUCTION

Datasets used to train machine learning (ML) classifiers often contain label noise due to various factors: (1) Non-stationary data, in which human-assigned labels cannot keep up with rapid data changes, (2) Inconsistent human annotations due to insufficient understanding of the task, and (3) Low-quality surrogate label generators that trade accuracy for speed with semi-automated methods, such as rule-based decision engines, Large Language models prompted in Zero/Few-shot mode for annotation, or, clustering-based bulk-labeling systems. Label noise makes it challenging for ML models to learn the true data distribution, resulting in reduced accuracy during audits [1].

To address the issue of label noise, we propose **R**obust **T**raining with **T**rust **S**cores (**RT2S**), a framework that trains a machine learning model using labels provided by a relatively low quality annotation system. RT2S computes a label quality score ("trust score") for every training sample, with higher scores indicating more reliable labels (see fig. 1). We ensemble the trust scores from two sources:

CIKM '23, October 21–25, 2023, Birmingham, United Kingdom

ACM ISBN 979-8-4007-0124-5/23/10.
https://doi.org/10.1145/3583780.3615996

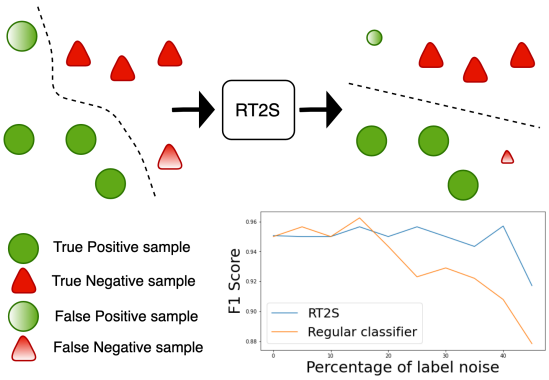

Figure 1: RT2S enables training a robust classifier by leveraging "trust scores". Noisy samples get low trust scores as opposed to clean ones. Samples with high trust scores carry more weights during training, and those with low trust scores have lesser impact (illustrated here by reduced size).

(1) the confidence on the observed label, given the out-of-fold prediction probabilities from a trained classifier, and (2) the likelihood of having the correct label estimated by a Large Language Model (LLM). Subsequently, the generated trust scores are used as sample weights to reduce the effect of noise in training data. RT2S falls under *Robust Training* [8], where the objective is to improve model performance when the labeled data is noisy. Robust Training approaches include: (A) Data cleaning, which can be achieved using methods such as k-nearest neighbor, outlier detection, or, state-of-the-art Confident Learning [6] etc. However, this approach can lead to over-cleaning by removing examples with true labels, and can have adverse effects on generalization capability for minority classes [7]. (B) Robust loss has also been shown to be effective for learning in the presence of noisy labels, but it can be slower to converge than cross-entropy loss due to gradient saturation [2]. (C) Probabilistic methods involve estimating the confidence of each label and using it to weigh the training samples [7, 8]. Although RT2S is conceptually similar to this approach, it differs in its methodology for generating label quality scores. RT2S employs LLMs to identify *biased* noise *i.e.* where annotators consistently misclassify data due to poor understanding of the classification rationale, and label quality scoring techniques from confident learning (CL) to detect *unbiased, class-conditional* noise.

In this work, we showcase the effectiveness of RT2S on six *binary* product classification tasks from our internal customer teams, wherein we employ noisy labels, generated by a rule-based classification system, as a proxy labeling technique to train an ML classifier (and no human annotations). Our experimental findings indicate that RT2S achieves higher accuracy compared to the other baselines consistently across all datasets.

2 RT2S

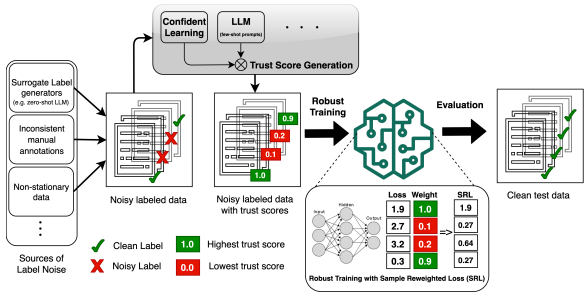

Figure 2: Label noise may come from various sources such as, surrogate label generators, inconsistent human labels *etc.* RT2S computes a quality score for every sample in the labeled data combining (1) Confident Learning based quality scores using out-of-fold predictions, (2) LLM based estimation of correct labels given task rationale and few-shot examples as context. RT2S uses trust scores as sample weights to train a classifier on a downstream task, and (optionally) optimize thresholds to deliver higher recall at a given precision target.

RT2S constitutes of two components: (1) Trust Score Generation, and (2) Robust Training with Sample Reweighted Loss (see fig. 2). For each training sample, we use the normalized margin formulation [4] to compute trust score, which measures the likelihood of its observed label being equal to its true (latent) label. The smallest normalized margin scores occur for samples whose observed label is different from the true label. RT2S currently leverages two models to independently estimate the probability distribution over all classes, and ensembles them to produce a trust score for each training sample. The first model follows the Confident Learning recipe, gets trained in a stratified K-fold fashion to generate out-of-fold scores. Inspired by the success of foundation models in annotation [3], we use an off-the-shelf LLM as our second model. We provide it with few-shot, in-context examples as part of the prompt, and generate a labeling decision independently for each training sample. Every product classification dataset also has a rationale that describes the intent of classification in natural language. We pass this as additional context to the LLM along with the instruction and in-context examples to generate a "yes"/"no" labeling decision, and then extract the normalized probability scores from the output. During training, we use trust scores as "sample weights" to minimize the expected Sample Reweighted Loss [5],

$$\mathcal{L}(f_\Theta; \tilde{S}) = \frac{1}{|\tilde{S}|} \sum_{(\mathbf{x},\tilde{\mathbf{y}})\in\tilde{S}} \underbrace{w_{\mathbf{x},\tilde{\mathbf{y}}} * \ell(f_\Theta(\mathbf{x}), \tilde{\mathbf{y}})}_{\text{Sample Reweighted Loss}}$$

where ℓ is the cross entropy loss. The parameters Θ of the model f is learnt iteratively using mini-batch gradient descent. Samples with high trust scores (weights close to 1) are expected to have clean labels. Hence, we penalize the model for incurring loss on such examples during training. On the contrary, samples with low trust scores (weights close to 0) are expected to have noisy labels, and we discount the network if it makes mistakes in predicting the given labels. This prevents the model from over-fitting to the noisy training set, and generalize well over the unseen data.

Table 1: Comparing *accuracy* (in %) of RT2S against the baselines on clean test sets from 6 binary product classification datasets. $P_{\tilde{S}}$ denotes the % of positive labels in the noisy training set $\tilde{S}$ ($10k$ samples), and P_T denotes the % of positive labels in clean test set T ($2k$ samples). For the ML models, accuracy is calculated based on *argmax* prediction.

Dataset	$P_{\tilde{S}}$	P_T	RB	Base-ML	CL	RT2S
Dataset A	69.42	86	81.61	81.58	83.61	**86.56**
Dataset B	45.32	59.75	75.34	77.11	77.72	**77.96**
Dataset C	65.74	73.84	79.64	79.47	80.37	**83.25**
Dataset D	77.16	84.1	87.68	88.41	89.68	**90.47**
Dataset E	90.8	71.3	82.12	82.82	82.81	**83.65**
Dataset F	22.63	51.65	65.83	68.65	70.19	**70.56**

3 RESULTS

We compare the performance of RT2S against 3 baselines on 6 *binary* product classification datasets having low-quality labels generated by a rule-based classifier. The baseline methods are: (a) Rule-based classifier **(RB)**, (b) Base Classifier **(Base-ML)** *i.e.* the classifier trained on noisy data with a default weight of 1, and (c) Confident Learning **(CL)** [6] the state-of-the-art data cleaning method for learning in the presence of label noise. Our ML classifier is an inference-optimized multi-lingual classifier called TinyM4 (a one-layer BERT model distilled from BERT-base and fine-tuned on our internal data). As per standard practice, we use a single linear layer on top of the [CLS] token embedding to obtain logits for the classification task. We evaluate all the methods (including RB) on a held-out clean test set which is further annotated by subject matter experts. We see that RT2S outperforms all baselines; an average accuracy improvement of +4.38% (max +7.18%) over the rule-based classifier. RT2S eliminates the need to train classifiers from scratch with human annotations, instead uses "pseudo labels" to balance the trade-off between speed and accuracy. In production, RT2S is helping internal customer teams to migrate existing rule-based product classifiers to ML-based product classifiers with +39% average improvement in recall, without compromising on precision.

REFERENCES

[1] Gorkem Algan and Ilkay Ulusoy. 2020. Label Noise Types and Their Effects on Deep Learning. *ArXiv* abs/2003.10471 (2020).
[2] Aritra Ghosh, Himanshu Kumar, and P. S. Sastry. 2017. Robust Loss Functions under Label Noise for Deep Neural Networks. arXiv:1712.09482 [stat.ML]
[3] Xingwei He, Zhenghao Lin, Yeyun Gong, A-Long Jin, Hang Zhang, Chen Lin, Jian Jiao, Siu Ming Yiu, Nan Duan, and Weizhu Chen. 2023. AnnoLLM: Making Large Language Models to Be Better Crowdsourced Annotators. arXiv:2303.16854 [cs.CL]
[4] Johnson Kuan and Jonas Mueller. 2022. Model-agnostic label quality scoring to detect real-world label errors. In *ICML DataPerf Workshop*.
[5] Tongliang Liu and Dacheng Tao. 2015. Classification with noisy labels by importance reweighting. *IEEE Transactions on pattern analysis and machine intelligence* 38, 3 (2015), 447–461.
[6] Curtis Northcutt, Lu Jiang, and Isaac Chuang. 2021. Confident learning: Estimating uncertainty in dataset labels. *Journal of Artificial Intelligence Research* 70 (2021), 1373–1411.
[7] Umaa Rebbapragada and Carla E. Brodley. 2007. Class Noise Mitigation Through Instance Weighting. In *Machine Learning: ECML 2007*, Joost N. Kok, Jacek Koronacki, Raomon Lopez de Mantaras, Stan Matwin, Dunja Mladenič, and Andrzej Skowron (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 708–715.
[8] Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. 2022. Learning from noisy labels with deep neural networks: A survey. *IEEE Transactions on Neural Networks and Learning Systems* (2022).