UoR at SemEval-2021 Task 12: On Crowd Annotations; Learning with Disagreements to Optimise Crowd Truth

Emmanuel Osei-Brefo, Thanet Markchom , Huizhi Liang
University of Reading
White Knights, Berkshire, RG6 6AH
United Kingdom
e.osei-brefo@pgr.reading.ac.uk, t.markchom@pgr.reading.ac.uk
huizhi.liang@reading.ac.uk

Abstract

Crowdsourcing has been ubiquitously used for annotating enormous collections of data. However, the major obstacles to using crowd-sourced labels are noise and errors from non-expert annotations. In this work, two approaches dealing with the noise and errors in crowd-sourced labels are proposed. The first approach uses Sharpness-Aware Minimization (SAM), an optimization technique robust to noisy labels. The other approach leverages a neural network layer called softmax-Crowdlayer specifically designed to learn from crowd-sourced annotations. According to the results, the proposed approaches can improve the performance of the Wide Residual Network model and Multi-layer Perception model applied on crowd-sourced datasets in the image processing domain. It also has similar and comparable results with the majority voting technique when applied to the sequential data domain whereby the Bidirectional Encoder Representations from Transformers (BERT) is used as the base model in both instances.

1 Introduction

In recent years, there has been some major advancement in the use of deep learning for solving artificial intelligence problems in different domains such as sentiment analysis, image classification, natural language inference, speech recognition object detection. They have also been used in many other numerous cases where human disagreements are encountered such as speech recognition, visual object recognition, object detection and machine translation (Rodrigues and Pereira, 2018). It is however, an essential requirement for deep learning models to utilise labelled data to undertake the representational learning of the underlying datasets. These labelled data are most at times not available and hence the need for humans to manually undertake the labelling of these data becomes a necessity.

In recent years, crowd-sourcing has been used in the annotation of large collections of data and has proven to be an efficient and cost-effective means of obtaining labeled data as compared to expert labelling (Snow et al., 2008)

It has been utilised in the generation of image annotations to train computer vision systems (Raykar et al., 2010), to provide the linguistic annotations used for Natural Language Processing (NLP) tasks (Snow et al., 2008), and has also been used to collect the relevant judgments needed to optimize search engines (Alonso, 2013) .

It is a well known fact that crowd-sourced labels are known to be associated with noise and errors as a result of the annotations being provided by annotators with uneven expertise and dedication which can result in the compromise of practical applications that uses such data (Zhang et al., 2016). This paper therefore seeks to apply a novel approach to minimize and mitigate the noise and errors in crowd sourced labels. The aim is to investigate the use of a unified testing framework to learn from disagreements using crowd source labels collected from different annotators.

2 Related Work

Crowdsourcing has proven to be an inexpensive and efficient way to collect large labels of data and has attracted much research interest from the machine learning community to address noise and unreliabilities associated with them. The proposal for using an Expected Maximization (EM) algorithm to obtain density estimate rate of errors of patients providing conflicting responses to medical questions by Dawid and Skene (1979), is one of the key pioneer contributions to this field. This work served as the catalyst for many other approaches used for the aggregation of labels from crowd annotators with different levels of expertise, such as

Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021), pages 1303–1309
Bangkok, Thailand (online), August 5–6, 2021.

the one proposed in Whitehill et al. (2009), which further extends Dawid and Skene's model by also accounting for item difficulty in the context of image classification. Similarly, Ipeirotis et al. (2010) proposed using Dawid and Skene's approach to extract a single quality score for each worker that low-quality workers to be pruned. The approach proposed in our paper contrast with this line of work, by allowing neural networks to be trained directly on the softmax output of the noisy labels of multiple annotators, thereby avoiding the need to resort to prior label aggregation schemes. Smyth et al. (1995) also collated the opinions of many experts to establish ground truth and there has been a large body of research work using EM approaches to annotate labels for datasets by many experts (Whitehill et al., 2009; Raykar and Yu, 2012).

Rodrigues et al. (2014) also used the EM approach of labelling datasets by experts through the use of Gaussian Process classifiers. Rodrigues and Pereira (2018) also deployed the use of crowd layer with a CNN model to capture the biases of different annotators and correct them, our approach is the first to be built on the Wide Residual Network (WideResNet) model (Zagoruyko and Komodakis, 2017) and the Bidirectional Encoder Representations from Transformers (BERT) model (Devlin et al., 2019). Our approach differs from the method used by Rodrigues and Pereira (2018) because our technique initially finds the softmax of the output of the crowd responses before it is used for the modelling whereas the Rodrigues and Pereira (2018) approach works on the responses from the crowd directly.

3 Systems Description

These systems are proposed for image classification tasks and NLP tasks with sub-task-specific modifications and training schemes applied to each of the dataset.

3.1 softmax-Crowdlayer

A special type of network layer known as softmax-Crowdlayer initially proposed by (Rodrigues and Pereira, 2018), was used to train a deep neural network directly from the noisy labels of multiple annotators from the crowd-sourced data. It used the output layer of a deep neural network as its input and was trained to learn from an annotator-specific mapping from the output layer to the labels of the different soft-maxed crowd annotators; and by so doing it was able to learn the reliability and biases of each annotator in the process. As can be seen from Figure 1, which is the generalised architecture encompassing either a Multi-layer Perceptron (MLP), WideResNet, or BERT as its' base model, was used together with a softmax-Crowdlayer for the respective datasets. The output layer from the deep neural network served as a bottleneck and input for the crowd Annotators to learn from. It used a specialised cross-entropy loss known as Masked Multi Cross Entropy loss during training to handle the missing answers from Annotators. After the training of the network with the crowd layer and the specialised loss function, the crowd layer was removed to expose the Bottleneck layer which was then used to make the predictions.

The intuition behind the deployment of the crowd layer on top of the base model was that; the softmax-Crowdlayer would adjust the gradients from the labels of each annotator depending on their level of expertism and adjusts their weights and propagate the errors through the entire neural network system.

Sections 3.2 covers the use of WideResNet together with SAM on the CIFAR10-IC dataset whilst sections 3.3 and 3.4 covers the use of softmax-Crowdlayer for image classification whilst section 3.5 explores the use of BERT and softmax-Crowdlayer to cover the NLP aspect of the task which has been visualised in figure 1. The motivation behind the preference of the BERT model over the baseline models was to investigate the potential of using BERT, which is a state-of-the-art model, with the softmax-Crowdlayer.

3.2 WideResNet with Sharpness Aware Minimisation (SAM) For Majority Voting

The CIFAR10 dataset had a model made with WideResNet; first implemented by (Zagoruyko and Komodakis, 2017). A widening factor of 12 and convolutions of size together with 16 layers were used. A learning rate of 0.1 with weight decay of 0.001 and momentum of 0.8 was used with the SAM optimiser which had schochatic Gradient Descent (SGD) as its base optimiser. The training epochs for the dataset were scheduled in batches of 1000 for 60, 5, 10 and 20 respectively. The minimization of the commonly used loss functions such cross-entropy and the use of the custom Masked loss function designed specifically for the crowd layer on the CIFAR10-IC were not sufficient to

achieve superior results since the training loss landscapes of models used for noisy labels are complex and non-convex, with a multiplicity of local and global minima (Foret et al., 2020).

The Sharpness-Aware Minimization (SAM) Foret et al. (2020), was applied to the CIFAR dataset with the use of WideResNet model generalization which aided in the simultaneous loss in value and sharpness of the noisy labels from the crowd annotators as it has been shown to be robust to noisy labels (Foret et al., 2020). The inner working of the sharpness Aware Minimization is such that rather than using a parameter value that simply have low training loss value, a parameter value whose entire neighborhoods have uniform training loss value is the utilised.

The SAM optimiser technique was not applied to the NLP tasks because, its performance on them was not as good as that of the CIFAR-10 dataset.

3.3 WideResNet with softmax-Crowdlayer for CIFAR10-IC Dataset

The CIFAR10-IC data was made up of transformed Images that belonged to one of the 10 classes below: 'plane', 'car', 'bird', 'cat', 'deer', 'dog', 'frog', 'horse', 'ship', 'truck'.

The WideResNet described in section 3.2 was used as the base model which had a softmax-Crowdlayer added to the output layer and through the action of back-propagation, it was able to correct the errors of the 2571 Annotators. A training epoch of 400 and batch size of 64 were used for with this approach. One hot encoding, together with a specialised function were used to generate the set of missing annotations which was then trained using the masked multi cross-entropy loss function for error corrections and predictions through the weights update.

3.4 MLP with softmax-Crowdlayer for LabelMe-IC Dataset

The LabelMe-IC data was made up of VGG16 encoded images that belonged to one of the 8 categories or classes below: 'highway', 'inside city', 'tall building', 'street', 'forest', 'coast', 'mountain' or 'open country'

This was an image classification task that had a standard MLP architecture together with softmax-Crowdlayer applied to it. The MLP was made up of 4 hidden layers with 128 Relu Units each, an optimiser made of Adam optimizer, loss function made of categorical cross entropy and a drop out of 0.2. A training epoch of 400 and batch size of 32 were used. The output layer had a softmax activation that outputted to the 8 distinct classes highlighted earlier. The softmax-Crowdlayer described in section 3.1 was then connected to this output layer where the Annotators errors and biases were back-propagated through a training scheme which reduced the noise in the crowd Annotators through the use of a specialised loss function to handle crowd annotations known as masked multi cross entropy loss function.

3.5 BERT with softmax-Crowdlayer for Gimpel-POS and PDIS Datasets

In Gimpel-POS dataset, each sample consisted of a tweeted text, a specific word/token appears in a tweeted text and a crowd label which is a list of multiple labels from different annotators. The task was to predict a part of speech (POS) of a given token. The POS labels include 'ADJ' (adjective), 'ADP' (adposition), 'ADV' (adverb), 'CCONJ' (coordinating conjunction), 'DET' (determiner), 'NOUN' (noun), 'NUM' (numeral), 'PRON' (pronoun), 'PART' (particle or other functional word), 'PUNCT' (punctuation), 'VERB' (verb) and 'X' (others). Table 1 shows an example of Gimpel-POS dataset. In this example, 'Texas' is a token needed to be tagged. It is at the beginning of the tweeted text shown in the first row. Considering the crowd label provided, the first and the second annotators both labeled this token as a noun, while the last annotator labeled this token as a pronoun.

Tweeted text	Texas Rangers are in the World Series! Go Rangers!
Token	Texas
Crowd label	[NOUN,NOUN,PRON]

Table 1: An example of Gimpel-POS dataset

Considering PDIS dataset, the goal was to predict whether a given noun phrase refers to new information or to old information in a document. Each sample consisted of a document (tokenised sentences), a noun phrase appear in the document, a pre-computed syntactic feature of a given noun phrase, and a crowd label. Table 2 shows an example of PDIS dataset. The document and the noun phrase are in the first and the second row of the table respectively. The noun phrase is 'The cat' at the beginning of the document. Syntactic feature

of this noun phrase is a feature vector shown in the third row. The fourth row shows a crowd label of the given noun phrase. The first and the second annotator labeled the noun phrase as 0 and 1 respectively 0 means that the noun phrase refers to new information and 1 means that it refers to old information.

Document	The cat ate the rat. Thereafter the dog ate the cat.
Noun phrase	The cat
Syntactic feature	[0,1,0,..,0]
Crowd label	[0,1]

Table 2: An example of PDIS dataset

In this work, we propose to fine-tune the pre-trained BERT model for both Gimpel-POS task and PDIS task based on crowd labels. To do so, the original input format of both tasks was firstly converted to the BERT conventional format. For each sample in Gimpel-POS dataset, a tweeted text and a given token were first concatenated in the following format:

[CLS] Tweeted text [SEP] Token [SEP]

where '[CLS]' token is added for classification and two '[SEP]' tokens are used to identify the boundary of a tweeted text and a token. Similarly, for PDIS dataset, a document is also concatenated with a noun phrase as follows:

[CLS] + Document + [SEP] + Noun phrase + [SEP]

These concatenated texts are used for fine-tuning the pre-trained BERT model.

To fine-tune the pre-trained BERT model, a dense layer was added at the end of the pre-trained BERT model. This layer took a '[CLS]' token embedding from the pre-trained BERT model as an input and outputted a vector with the size equal to the number of classes in either dataset (12 for Gimpel-POS and 2 for PDIS). A softmax activation layer was added after the dense layer to compute the probabilities of each class. These additional layers can be seen as a classifier module that is added on top of the pre-trained BERT model. This is a common way to fine-tune the pre-trained BERT model for a specific task with regular labels as targets (Devlin et al., 2019).

In order to deal with crowd labels in the datasets, the softmax-Crowdlayer was added next to the classifier module. Similarly to the MLP model with the crowd layer highlighted in 3.4, The proposed model for fine-tuning the pre-trained BERT with the softmax-Crowdlayer is illustrated in Figure 1. The Gimpel-POS example in 1 is used for demonstration in this figure. As previously mentioned, only the '[CLS]' token is passed through the additional classifier module to predict primary classification output. This output is further used as an input of the softmax-Crowdlayer to predict the final output as described in the previous section. The proposed model can be instantly applied with PDIS dataset by changing the output size of the dense layer in the classifier module to 2. Due to lack of resources, the fine tuning of all the Bert model was run for 1 epoch.

4 Results and Discussion

The results were evaluated using two metrics known as $F1$ score, referred to as hard evaluation and and cross Entropy, referred to as soft evaluation. Models with Higher $F1$ scores and lower cross entropy values are the desired outcomes expected from the models.

As can be seen in Table 3, The use of MLP together with the softmax-Crowdlayer on the LabelMe-IC dataset achieved the highest $F1$ score of 0.7839, which was 0.739 greater than the majority voting model provided as the baseline model by the task organisers and also had a comparative lowest cross entropy value of 1.7693. The vast difference in the performance of the majority voting and the softmax-Crowdlayer can be attributed to the calculation of the number of missing annotations together with the ability of the softmax-Crowdlayer to learn the true labels from the crowd labels. This leads to the correction of the errors and mislabelling from inexperienced annotators through the process of back-propagation. The majority voting does not have this unique ability and therefore uses the wrong labelling without any of such adjustments.

The use of WideResNet together with SAM resulted in a superior performance with $F1$ score of 0.7693 and cross entropy of 0.8274 as compared to the performance WideResnet with the softmax-Crowdlayer which had an $F1$ score of 0.4427 and cross entropy of 1.9286 when applied to the CIFAR10-IC. It's cross entropy of 1.9286 was better than the baseline majority method which was 2.8306. The PDIS data which was fine-tuned with a pre-trained BERT model plus softmax-crowdlayer

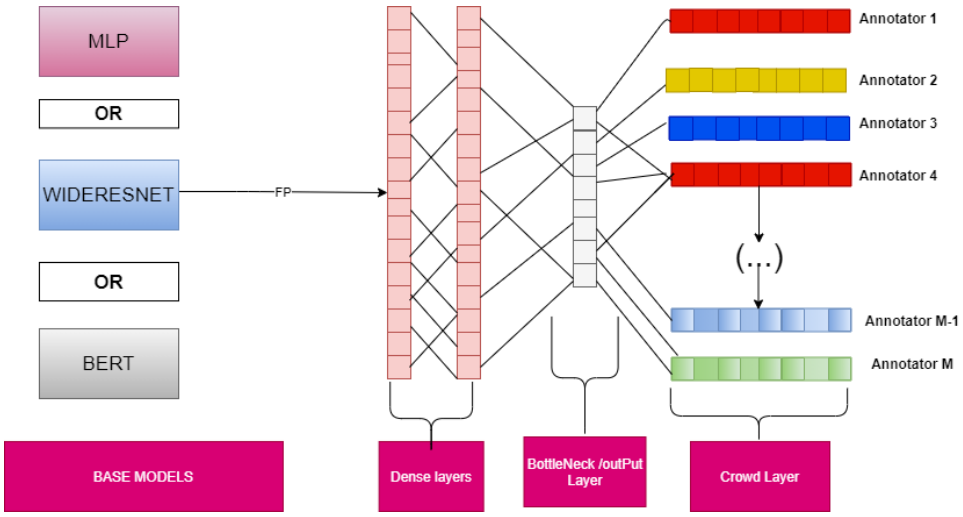


Figure 1: The proposed softmax-Crowdlayer on top of the respective Base Models

had $F1$ score of 0.4379 and cross entropy of 0.8295. The BERT + softmax-Crowdlayer did not perform comparatively well when applied to the Gimpel-POS data since it only managed to achieve an $F1$ score of 0.1254 and corresponding cross entropy of 2.3318. From Table 3 it can also be seen that the BERT + Majority voting had the same results as the BERT + softmax-Crowdlayer model so further investigation needs to be conducted to find out why this was so. As can be seen in Table 3, the use of the full base model provided for the PDIS and Gimpel-POS by the organisers achieved superior results and should have been used with the softmax-crowdlayer, but it could not be done because the full base model provided by the organisers was written in Pytorch framework whilst the softmax-crowdlawyer was written in Keras. There should therefore have been the need to convert the full base model to Keras before using the softmax-crowdlayer and it's eventual evaluation, but as a result of the limited availability of time, it has been reserved as part of our future work to be covered in section 5

Refer to Appendix A for the analysis of the class distribution of the datasets.

5 Conclusion

This paper used a softmax-Crowdlayer approach combined with a deep neural network to train noisy labels from multiple crowd annotators. WideResNet together with softmax-Crowdlayer has been applied on CIFAR10-IC datasets, whilst MLP combined with softmax-Crowdlayer has been used on the LabelMe-IC data and BERT combined with softmax-Crowdlayer has been used on Gimpel-POS and PDIS data respectively.

Future work will explore the effect of the distribution of the class annotation on the labeling accuracy and also investigate more efficient approaches of combining the BERT model with the softmax-Crowdlayer to further improve the results. It will also involve the application of the softmax-crowdlayer on the Humour dataset which was not included in this work due to time constraint posed as a result of the complicated data points of the humour dataset.

References

Omar Alonso. 2013. Implementing crowdsourcing-based relevance experimentation: an industrial perspective. *Information Retrieval*, 16(2):101–120.

A. P. Dawid and A. M. Skene. 1979. Maximum likelihood estimation of observer error-rates using the em algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):20–28.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. 2020. Sharpness-aware minimization for efficiently improving generalization.

Panagiotis G. Ipeirotis, Foster Provost, and Jing Wang. 2010. Quality management on amazon mechanical turk. In *Proceedings of the Acm sigkdd Workshop on Human Computation*, hcomp '10, page 64–67, New York, NY, USA. Association for Computing Machinery.

SubTask	Model	F1 Score	Cross Entropy
CIFAR10-IC	WideResNet + Majority Voting+SAM	**0.7693**	**0.8274**
CIFAR10-IC	WideResNet + Majority Voting (NO SAM)	0.7156	1.1163
CIFAR10-IC	WideResNet + softmax-Crowdlayer	0.4427	1.9286
CIFAR10-IC	ResNet + Majority Voting	0.6306	2.8306
LabelMe-IC	MLP + softmax-Crowdlayer	**0.7839**	**1.7693**
LabelMe-IC	MLP + Majority Voting	0.0449	2.2100
PDIS	BERT + softmax-Crowdlayer	0.4739	0.8295
PDIS	BERT + Majority Voting	0.4739	0.6987
PDIS	CharCNN + Majority Voting	0.5611	0.6892
Gimpel-POS	BERT + softmax-Crowdlayer	0.1254	2.3318
Gimpel-POS	BERT + Majority Voting	0.1254	2.257
Gimpel-POS	RNN + Majority Voting	0.7626	1.0884

Table 3: Evaluation results

Vikas C. Raykar and Shipeng Yu. 2012. Eliminating spammers and ranking annotators for crowdsourced labeling tasks. *J. Mach. Learn. Res.*, 13(null):491–518.

Vikas C. Raykar, Shipeng Yu, Linda H. Zhao, Gerardo Hermosillo Valadez, Charles Florin, Luca Bogoni, and Linda Moy. 2010. Learning from crowds. *Journal of Machine Learning Research*, 11(43):1297–1322.

Filipe Rodrigues and Francisco Pereira. 2018. Deep learning from crowds.

Filipe Rodrigues, Francisco Pereira, and Bernardete Ribeiro. 2014. Gaussian process classification and active learning with multiple annotators. In *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 433–441, Bejing, China. PMLR.

Padhraic Smyth, Usama Fayyad, Michael Burl, Pietro Perona, and Pierre Baldi. 1995. Inferring ground truth from subjective labelling of venus images. In *Advances in Neural Information Processing Systems*, volume 7. MIT Press.

Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Ng. 2008. Cheap and fast – but is it good? evaluating non-expert annotations for natural language tasks. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 254–263, Honolulu, Hawaii. Association for Computational Linguistics.

Jacob Whitehill, Ting-fan Wu, Jacob Bergsma, Javier Movellan, and Paul Ruvolo. 2009. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. In *Advances in Neural Information Processing Systems*, volume 22. Curran Associates, Inc.

Sergey Zagoruyko and Nikos Komodakis. 2017. Wide residual networks.

J. Zhang, X. Wu, and V. Sheng. 2016. Learning from crowdsourced labeled data: a survey. *Artificial Intelligence Review*, 46:543–576.

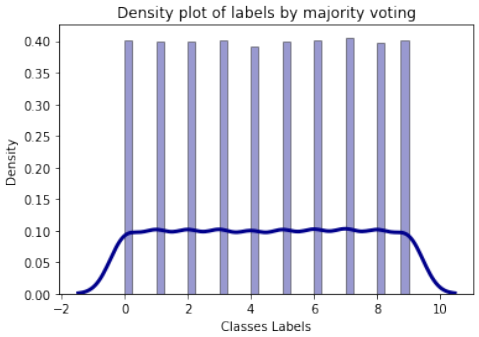
A Class label distribution analysis

The table 4, summarises the number of Annotators, number of data points, and number classes for each data set used. The Figure 2 contains the probability density estimates of how the annotators perceived the class labels to which each respective item belonged to. Based on the simple majority voting, it can be observed from Figure 2(a) that the distribution was uniform across all classes with the labeling ; **[0:airplane, 1:automobile, 2:bird, 3:cat, 4:deer, 5:dog, 6:frog, 7:horse, 8:ship, and 9:truck]** for the CIFAR10-IC dataset.

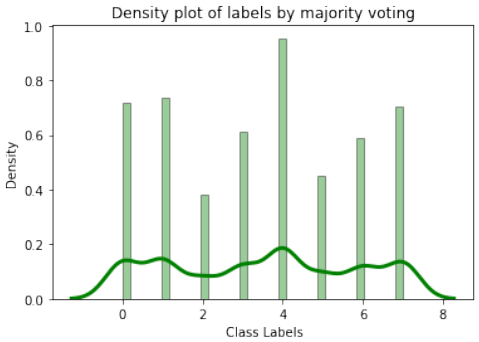
Figure 2(b) depicting the kernel density estimates of the LabelMe-IC data, captures the distribution of how the annotators labelled the data into their respective classes. Majority of the samples were labelled as forest with the respective encoding of the labels shown as; **[0:highway, 1:inside city, 2:tall building, 3:street, 4:forest, 5:coast, 6:mountain, 7:open country]**.

The distribution of the Gimpel-POS and PDIS datasets are represented in figures 2(c) and (d) respectively whose encoded labels have been provided earlier in section 3.5. The encoding for the Gimpel-POS class is shown as **[0:ADJ, 1:ADP, 2:ADV, 3:CCONJ, 4:DET, 5:NOUN, 6:NUM, 7:PRON, 8:PART, 9:PUNCT, 10:VERB, 11:X]**.

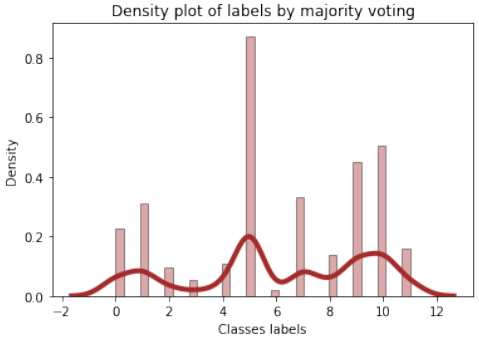
The labels of PDIS dataset are encoded as **[0: refer to new information, 1: refer to old information]**.



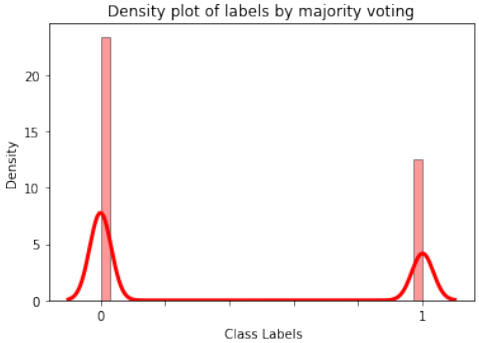
(a)



(b)



(c)



(d)

Figure 2: The kernel density plot of the distribution of the crowd labels by Annotators for the (a) CIFAR10-IC, (b) LabelMe-IC, (c) Gimpel-POS and (d) PDIS datasets

Dataset	#annotators	#classes	size
CIFAR10-IC	2571	10	7000
LabelMe-IC	59	8	5000
Gimpel-POS	177	12	8310
PDIS	1728	2	86936

Table 4: Summary statistics for each dataset used