Multi-View Knowledge Distillation from Crowd Annotations for Out-of-Domain Generalization

Anonymous ACL submission

Abstract

Selecting an effective training signal for tasks in natural language processing is difficult: expert annotations are expensive, and crowd-sourced annotations may not be reliable. At the same time, recent work in NLP has demonstrated that learning from a distribution over labels acquired from crowd annotations can be effective. However, the best method for acquiring these soft labels is inconsistent across tasks. This paper systematically analyzes this in the out-of-domain setting, adding to the NLP literature which has focused on in-domain evaluation, and proposes new methods for acquiring soft-labels from crowd-annotations by aggregating the distributions produced by existing methods. In particular, we propose to aggregate multiple-views of crowd annotations via temperature scaling and finding their Jensen-Shannon centroid. We demonstrate that these aggregation methods lead to best or near-best performance across four NLP tasks on out-of-domain test sets, mitigating fluctuations in performance when using the individual distributions. Additionally, aggregation results in best or near-best uncertainty estimation. We argue that aggregating different views of crowd-annotations is an effective way to ensure performance which is as good or better than the best individual view, which is useful given the inconsistency in performance of the individual methods.

1 Introduction

One of the primary concerns in supervised machine learning is how to define, collect, and use labels as training data for a given task. There are a multitude of tradeoffs associated with this decision, including the cost, the number of labels to collect, the time to collect those labels, the accuracy of those labels with respect to the task under consideration, and how well those labels enable model generalization. These tradeoffs are made based on how the labels are collected (e.g. crowdsourcing, expert labeling, distant supervision) and how they are trained on in practice, for example as one-hot categorical labels (hard labeling) or as a distribution over possible classes (soft labeling).

A large body of literature exists which examines all facets of this question (Uma et al., 2021). Recent work has used soft-labeling schemes for classification tasks as a method for improving both model accuracy and uncertainty estimation (Peterson et al., 2019; Uma et al., 2020; Fornaciari et al., 2021). When using soft-labels, models are trained to minimize the divergence between their predictive distribution and a distribution over the labels obtained from crowd annotations (Uma et al., 2020). While this has been shown to potentially improve model generalization for vision tasks (Peterson et al., 2019), little work has systematically compared how different soft-labeling schemes affect out-of-distribution performance and uncertainty estimation in NLP. We seek to fill this gap in this work, providing an in-depth study into soft-labeling techniques and best practices for improving model generalization and uncertainty estimation across eight methods, 4 NLP tasks, and 7 datasets.

Soft-labeling methods have been compared in both Fornaciari et al. (2021) and Uma et al. (2021) for an in-domain testing setting. These studies are primarily focused on identifying the best methods for learning from these soft distributions within a particular domain without going in great depth about which methods for obtaining soft-labels lead to best performance. As such, no clear best method emerges when comparing across soft-labeling approaches in the in-domain setting, making it difficult to decide which technique to use for a given task. Additionally, these studies do not examine the out of domain test setting, where the benefits of soft-labeling have been indicated in the computer vision literature (Peterson et al., 2019). Here, we demonstrate that aggregating soft-labels from different techniques into a single distribution offers more consistent performance across tasks. We therefore propose four multi-view aggregation methods to generate aggregated soft-labels, including three novel methods based on the Jensen-Shannon centroid and temperature scaling.

In sum, we make the following contributions:

1) A comparison of soft-labeling techniques for learning from crowd annotations for 4 NLP tasks across 7 datasets in the out-of-domain test setting, including text classification (recognizing textual entailment, medical relation extraction, and toxicity detection) and sequence tagging (part-of-speech tagging).
2) Novel methods for aggregating different views of soft-labels derived from crowd-annotations.
3) Insights and suggestions into best practices for different soft-labeling methods in terms of performance and uncertainty estimation.

2 Related Work

Learning from Crowd-Sourced Labels An efficient way to collect training data for a new task is to ask crowd annotators on platforms such as Amazon Mechanical Turk to manually annotate training data. How to select an appropriate training signal from these noisy crowd labels has a rich set of literature (e.g. see the survey from Paun et al. 2018). Many of these studies focus on Bayesian methods to learn a latent distribution over the true class for each sample, influenced by factors such as annotator behavior (Hovy et al., 2013; Dawid and Skene, 1979) and item difficulty (Carpenter, 2008), and selecting the mean of this distribution as the final label. However, selecting a single true label discards potentially useful information regarding uncertainty over classes inherent in many tasks, for example where items can be especially difficult or ambiguous (Gordon et al., 2021). Recent work has looked into how to learn directly from crowd-annotations (Uma et al., 2021). The work of Peterson et al. (2019) demonstrated that learning directly from crowd annotations treated as soft-labels using the softmax function leads to better out of distribution performance in computer vision. This line of work has been followed by Uma et al. (2020) and Fornaciari et al. (2021) in NLP, looking at the use of the KL divergence as an effective loss. The survey of Uma et al. (2021) provides an extensive set of experiments comparing methods for learning from crowd labels. What has not been done is a systematic comparison of different soft-labeling methods in the out of domain setting. We fill this gap in this work, and propose new methods for aggregating soft-labels which yield more consistent and robust performance than previous methods without requiring new annotations or learning methods.

Knowledge Distillation Knowledge distillation seeks to build compact but robust models by training them on the probability distribution learned by a much larger teacher network (Ba and Caruana, 2014; Hinton et al., 2015). The goal is to impart the “dark knowledge” contained in the distribution learned by the larger network, which can indicate similarities between features and classes if the output from the classifier is well calibrated (e.g. via temperature scaling (Hinton et al., 2015) or ensembling (Hinton et al., 2015; Allen-Zhu and Li, 2020)). Allen-Zhu and Li (2020) demonstrate that when distilling from an ensemble, the data used to train the ensemble should constitute a multi-view structure (i.e. multiple different features in the data are predictive of a particular class) for best performance. Inspired by this, we develop several methods for aggregating multiple views of crowd-sourced labels in order to obtain a distribution that can induce robust classifiers in the out-of-domain setting. “Multi-view” in this work is defined as multiple distributions from crowd annotations that are explained by different factors e.g. annotator behavior or raw number of votes per class.

3 Methods

We build upon a rich literature around the topic of learning from crowd annotations, particularly on learning from soft-targets: distributions over classes obtained from annotations as opposed to selecting a single hard label. In this, samples have their probability mass distributed over multiple classes, which can help regularize a downstream classifier and reflect potential “dark knowledge” (Hinton et al., 2015) learnable from the crowd annotations. Multiple soft-labeling methods have been demonstrated to provide good training signals on different NLP tasks, but none of these methods are consistently best across tasks (Uma et al., 2021). Given this, we start with several well-studied methods for learning from crowd-labels, described in Section 3.1 (Uma et al., 2020; Fornaciari et al., 2021; Hovy et al., 2013; Dawid and Skene, 1979), adding to this literature by analyzing their performance when considering generalization to out of domain data. Then, we propose several methods for aggregating these distributions in Section 3.2, which we will demonstrate lead to consistent performance across tasks.

3.1 Soft Labeling Methods

We experiment with four widely used methods for obtaining soft labels: two based on normalizing over annotations counts and two based on Bayesian models. More detailed descriptions of these methods are given in Appendix A.

- Standard normalization: Transforms a set of crowd labels into a distribution by averaging the number of votes given to a particular label by the total number of annotations on that item (Uma et al., 2020).
- Softmax normalization: Instead of directly averaging over the number of annotations for a given item, take a softmax over the votes. This ensures that some probability mass is distributed to each label for each sample (Peterson et al., 2019; Fornaciari et al., 2021).
- Dawid & Skene: The Bayesian model from Dawid and Skene (1979) which learns a posterior distribution over the true class for each sample based on each annotator's ability to correctly identify true instances of a given class.
- MACE: Multi-Annotator Competence Estimation (Hovy et al., 2013), another Bayesian model which models whether or not annotators are faithfully annotating each item or following a local spamming strategy which does not reflect the true underlying label.

3.2 Combining Soft Labels

Each of the above methods will produce a distribution over labels which can be used in training; however different methods produce better training signals depending on the task (Uma et al., 2021). In order to acquire labels which capture the multiple views of the annotations learned by these methods, we develop novel methods for aggregating their soft labels. This is inexpensive, requiring zero additional annotations, and we will demonstrate that it is robust across tasks.

The goal for a single example x_i is as follows: given a set of categorical distributions $p_m(y_i|x_i)$ with $m \in \{1...M\}$ for M different distributions, produce a categorical distribution $p(y_i|x_i) = f(p_{1:M}(y_i|x_i))$ which will serve as a soft target for example x_i. Our hypothesis is that combining several different models (i.e. different **views** of the crowd-sourced annotations) will yield labels that can induce more robust classifiers as they will capture the uncertainty present in each of the individual distributions which are based on different factors (e.g. annotator behavior and raw class votes).

Averaging The most basic model to acquire an aggregated probability distribution is to take an average of the individual probabilities $p_{1:M}$. More formally, the averaging function f_a is:

$$f_a(p_{1:M}(y_i|x_i)) = \frac{1}{M}\sum_m p_m(y_i|x_i) \qquad (1)$$

This yields a distribution which is the center of mass of the given distributions $p_{1:M}$.

Jensen-Shannon Centroid The Jensen-Shannon centroid (JSC) is the minimizer of the sum of the Jensen-Shannon divergences (JSD) between a proposed distribution Q and a set of probability distributions $p_{1:M}$. It is defined as:

$$f_c(p_{1:M}(y_i|x_i)) = \arg\min_Q \sum_m \mathrm{JS}(p_m \| Q) \qquad (2)$$

where $\mathrm{JS}(P\|Q)$ is the JSD, a symmetric version of the Kullback-Leibler divergence (KLD), defined as follows for discrete probability distributions:

$$\mathrm{JS}(P\|Q) = \frac{1}{2}\mathrm{KLD}(P\|S) + \frac{1}{2}\mathrm{KLD}(Q\|S) \qquad (3)$$

$$S = \frac{1}{2}(P+Q)$$

$$\mathrm{KLD}(P\|Q) = \sum_j P^{(j)} \log \frac{P^{(j)}}{Q^{(j)}} \qquad (4)$$

Our hypothesis is that the JSC, unlike simple averaging, will be less influenced by highly disparate distributions in the ensemble which could negatively influence performance. To find the JSC, we use the ConCave-Convex procedure (CCCP, Yuille and Rangarajan 2001) developed in Nielsen (2020). The full derivation and definition of the method can be found in Nielsen (2020), Equations 94-104 and Algorithm 1, and a high level overview is given here in Appendix E.

Temperature Scaling One approach in knowledge distillation is to scale the softmax output of the larger teacher network prior to using it to produce soft labels to teach the smaller student network. Here, we develop a method for optimizing a temperature parameter for each distribution in our ensemble based on the JSD between distributions.

For each soft-labeling method p_m, we optimize a temperature parameter $T_m, m \in \{1...M\}$ which softens each distribution produced by that method. In other words, we produce softened distributions $\tilde{p}_m$ as:

$$\tilde{p}_m(y_i|x_i) = \text{softmax}(\frac{l_m(y_i|x_i)}{T_m}) \quad (5)$$

where l_m are the log-probabilities for a given sample. The temperature T_m is then optimized to minimize the JSD between each of the $\frac{M(M-1)}{2}$ combinations of distributions in the ensemble, the idea being to calibrate each distribution based on the uncertainty captured by each other distribution in the ensemble. Since optimizing this loss directly will encourage the temperature to scale to infinity, as the loss will be 0 when a large enough temperature drives all distributions to be uniform, we also add a regularization loss on the temperature parameters in order to discourage them from being exceedingly large. The final loss (assuming averaging the JSD over a batch of samples) is given in Equation 6.

$$\mathcal{L} = \frac{1}{Z}\sum_{j=1}^{M}\sum_{k=j+1}^{M} \text{JSD}(\tilde{p}_j \| \tilde{p}_k) + \lambda T_j^2 \quad (6)$$

where λ is a regularization constant and $Z = \frac{M(M-1)}{2}$. Finally, after optimizing for the temperature parameters T_m, we aggregate the distributions by averaging over the temperature scaled ensemble.

$$f_t(p_{1:M}) = f_a(\tilde{p}_{1:M}) \quad (7)$$

Hybrid Finally, we develop a hybrid approach where we first temperature scale the distributions in the ensemble via Equation 6, followed by finding the JSC as in Equation 2.

$$f_h(p_{1:M}) = f_c(\tilde{p}_{1:M}) \quad (8)$$

4 Experimental Setup

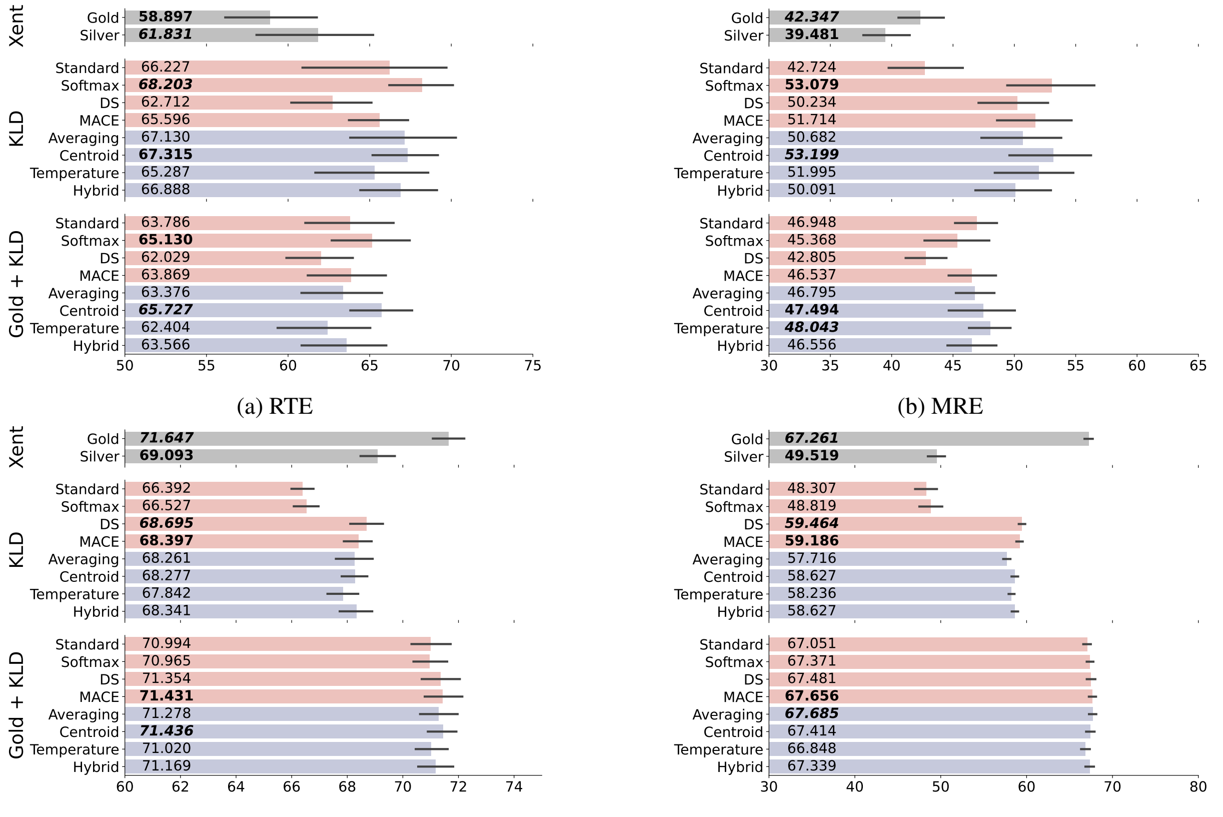
Our experiments serve to answer the following research questions:

- **RQ1**: Which methods for learning from crowd-sourced labels are most robust in out-of-domain settings?
- **RQ2**: Does aggregating multiple views of crowd annotations lead to more robust out-of-domain performance?
- **RQ3**: Which soft-labeling methods lead to better uncertainty estimation?

Our experiments focus on the out-of-domain setting. We use pairs of datasets which capture the same high-level tasks and where the training data has both gold and crowd-annotations available while the testing data only has gold annotations. We use dataset pairs with one of two sources of domain shift: 1) input data sourced from different corpora; 2) labels acquired from different sources. Additionally, two of our experiments have training sets with less than 1,000 samples. This setup lets us understand the impact of learning from crowd-labels on model generalization, whereas in the in-domain setting where train and test data use gold labels obtained from the same source, performance is dominated by the use of gold labels.

For all experiments we use RoBERTa as our base network (Liu et al., 2019) with the same training hyperparameters in order to provide a stable comparison across different soft-labeling techniques. Additionally, this allows us to observe how the same soft-labeling techniques on the same network perform on different tasks. For the soft-labeling experiments (labeled "KLD") we only use soft labels obtained using one of the crowd-labeling methods described in Section 3.1 and Section 3.2 and train using the KL divergence as the loss (as in previous work (Uma et al., 2021)). Additionally, we experiment with the multi-task learning setup used in Fornaciari et al. (2021) and Uma et al. (2021), where the model is trained on both gold labels and soft-targets (labeled "Gold + KLD"). This allows us to differentiate performance between when gold annotations are available vs. not, which is clearly beneficial in the in-domain test setting where the same method of acquiring gold labels is used for test data (Fornaciari et al., 2021), but not necessarily in the out-of-domain setting (Peterson et al., 2019). The tasks and datasets used in our experiments are described in the following paragraphs (full descriptions in Appendix B).

Recognizing Textual Entailment (RTE) The first task we consider is recognizing textual entailment (RTE). In the RTE task, a model must predict whether a hypothesis is entailed (i.e. supported) by a given premise. For training, we use the Pascal RTE-1 dataset (Dagan et al., 2005) with crowd-sourced labels from Snow et al. (2008) and for test we use the Stanford Natural Langauge Inference dataset (SNLI, Bowman et al. (2015)).



(a) RTE

(b) MRE

(c) POS

(d) Toxicity

Figure 1: F1 scores averaged across 20 random seeds. Grey are hard labels only, red are individual methods and blue are aggregation methods. Best results within each setting are given in ***bold italics***, second best results in **bold**.

Medical Relation Extraction (MRE) Medical relation extraction (MRE) seeks to predict what relations hold between biomedical entities in sentences from biomedical papers. The MRE dataset used for training in this work is the crowd-sourced dataset from (Dumitrache et al., 2018), focusing on the 975 sentence subset which received expert annotations specifically for the "cause" relationship (following previous work (Uma et al., 2021)). For test, we use the causal claim-strength dataset curated from (Wright and Augenstein, 2021), which contains 1,126 sentences from news articles and scientific papers related to health science labeled for causal claim strength and turned into a binary prediction task ("cause" and "not cause").

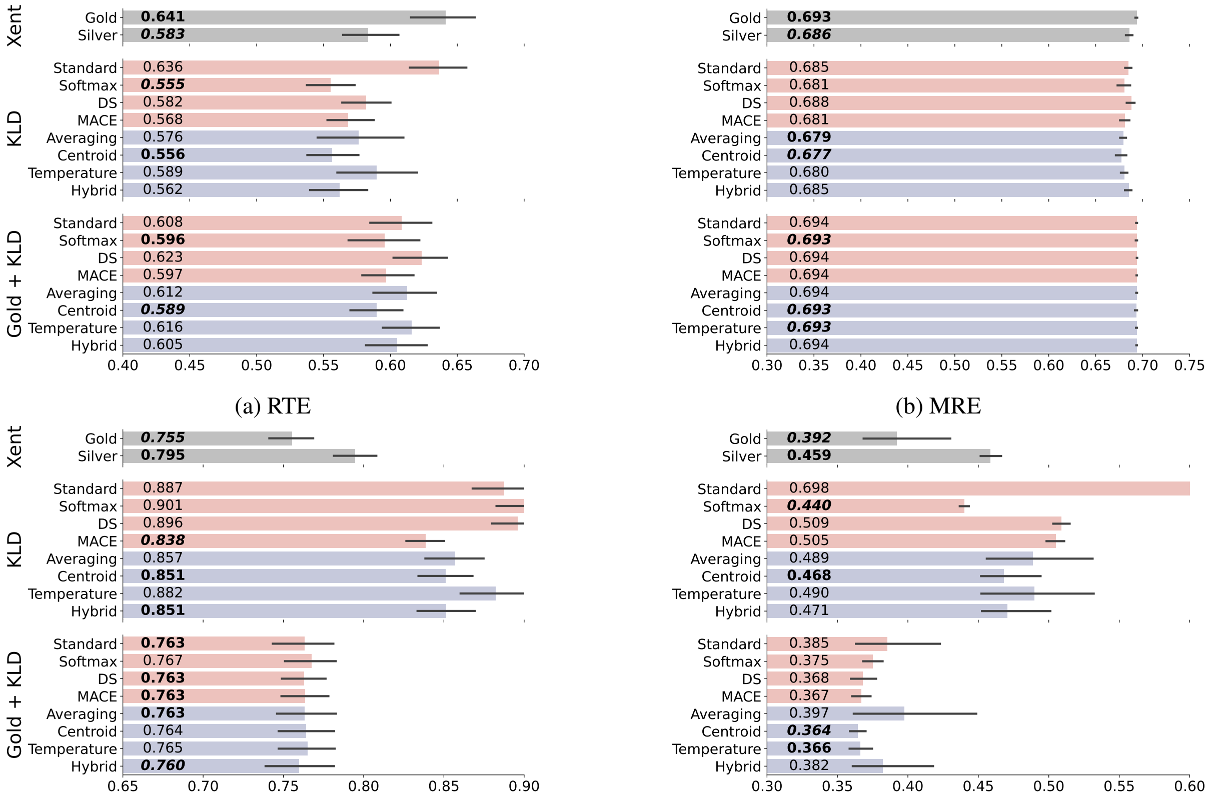
Part-of-Speech Tagging (POS) The POS tagging task is a sequence tagging task to predict the correct part-of-speech for each token in a sentence. For training data, we use the Gimpel dataset from Gimpel et al. (2011) with the crowd-sourced labels provided by Hovy et al. (2014); Plank et al. (2014). We use the publicly available sample of the Penn Treebank POS dataset (Marcus et al., 1993) accessed from NLTK (Bird, 2006) as our out-of-domain test set, which consists of 3,914 sentences from Wall Street Journal articles (100,676 tokens).

Toxicity Detection Finally, to measure performance on a highly subjective task, we use the toxicity detection dataset created as a part of the Google Jigsaw unintended bias in toxicity classification competition.[1] The dataset we use comes from Goyal et al. (2022), which annotated 25,500 comments from the original Civil Comments dataset. The pool of annotators is specifically selected and split into multiple rating pools based on self-indicated identity group membership (African American and LGBTQ). We randomly split the dataset into training and test, and for the test data we use the annotations in the original crowd-sourcing task; in other words, using a completely separate annotator pool that is not selected based on identity groups.

5 Results and Discussion

We evaluate the performance of each soft-labeling method across two metrics: F1 score and calibrated

[1] https://www.kaggle.com/competitions/jigsaw-unintended-bias-in-toxicity-classification



(a) RTE

(b) MRE

(c) POS

(d) Toxicity

Figure 2: Calibrated log-likelihood (CLL, ↓ better) averaged across 20 random seeds. Grey are hard labels only, red are individual methods and blue are aggregation methods. Best results within each setting are given in ***bold italics***, second best results in **bold**.

log-likelihood (CLL, Ashukha et al. 2020). Formal definitions of each metric can be found in Appendix C. Additionally, we show results using only gold labels (Gold) and only majority vote (Silver). We first discuss general observations from our results, and based on this provide answers for the research questions proposed in Section 4.

5.1 Raw Performance

Raw performance in terms of (macro) F1 score is shown in Figure 1.

Overall We see that the RTE and MRE datasets are much more difficult to generalize from than the POS and Jigsaw tasks, as reflected in the wide confidence intervals of the results. Additionally, gold labels in these two settings yield worse performance than simply training on soft labels, as opposed to the in-domain setting reported in Uma et al. (2021) where gold labels are needed for high performance. POS tagging sees the best performance when using only gold labels, contrasting with results reported in Uma et al. (2021) which show that adding soft labels with gold labels improves performance in the in-domain setting. For the Toxicity task, using out of domain hard labels (Silver) clearly leads to worse performance than using the original annotations (almost 20 F1 points drop). Using soft labels performs much better than this, with the aggregation methods being robust to very poor distributions. Additionally, augmenting gold labels with soft-labels obtained from new annotators still has benefits on this task.

Soft Labels Looking towards which soft-labeling method provides the best performance in the absence of gold labels, it is inconsistent across tasks. This was also seen in the survey by Uma et al. (2021). However, the aggregation methods are more **consistent** than the individual methods. In particular, the aggregation method using the JSC (Centroid in Figure 1) yields best or near-best performance across tasks, while the hybrid method works slightly better on POS tagging. This is despite fluctuations in performance for the individual methods across tasks. For example, the softmax method works well for RTE and MRE, but worse for POS tagging and much worse for Toxicity detection. The Bayesian methods show the opposite behavior, working well for POS tagging and Toxicity detection (potentially due to there being far more annotations from which to learn), but much worse for RTE and MRE. Aggregation is also resistant to low-performing individual distributions, as can be seen in the toxicity experiment where both the standard and softmax distributed labels produce significantly worse classifiers than those trained on labels from either Bayesian method, while each aggregation method remains close to the best performers. Finally, we also see that temperature scaling does not benefit performance in this setting, and robust performance is achieved with the JSC alone.

Gold + KLD Adding gold labels for the RTE and MRE tasks leads to worse performance, potentially due to the limited amount of labeled data. This adds further evidence to the literature that soft labels can provide benefits over gold labels for out-of-domain performance (Peterson et al., 2019). In terms of raw performance, gold labels are sufficient to obtain best performance for POS tagging, with soft labels not conferring benefits in the out-of-domain setting. This may be explained by the observation that the gold annotations for the POS dataset (Gimpel et al., 2011) were collected by researchers correcting labels for tweets pre-tagged by a tagger trained on Wall Street Journal articles (as in PTB), while the crowd-sourced annotations we use from (Hovy et al., 2014) are annotated from scratch with minimal context, only seeing three words at a time. As such, while there is a significant difference between the source of input data between train and test, there may be less difference in terms of gold labels. For the toxicity detection task, all methods perform within reasonable ranges of each other, with the Bayesian methods and basic averaging conferring slightly better performance.

5.2 Uncertainty Estimation

Uncertainty estimation in terms of CLL for each method and dataset can be seen in Figure 2.

Overall We see that uncertainty estimation as measured using CLL can be improved with the addition of soft-labels in all cases except for POS tagging. The benefits are again more pronounced for the RTE and MRE tasks, where training data is limited. We also see inconsistency from the individual soft labeling methods across tasks, while the aggregation methods (and particularly the JSC) offer much more consistent uncertainty estimation which is better or approximately equal to the performance of the best performing individual method.

Method	RTE	MRE	POS	Toxicity
Standard	0.919	0.764	0.799	0.784
Softmax	0.919	0.764	0.799	0.784
MACE	0.926	0.765	0.799	0.731
D&S	0.927	0.760	0.779	0.733
Average	0.927	0.765	0.799	0.754
Centroid	0.927	0.765	0.799	0.754
Temperature	0.930	0.766	0.799	0.757
Hybrid	0.930	0.765	0.799	0.757

Table 1: The accuracy of each annotation method with respect to the gold annotations in each dataset.

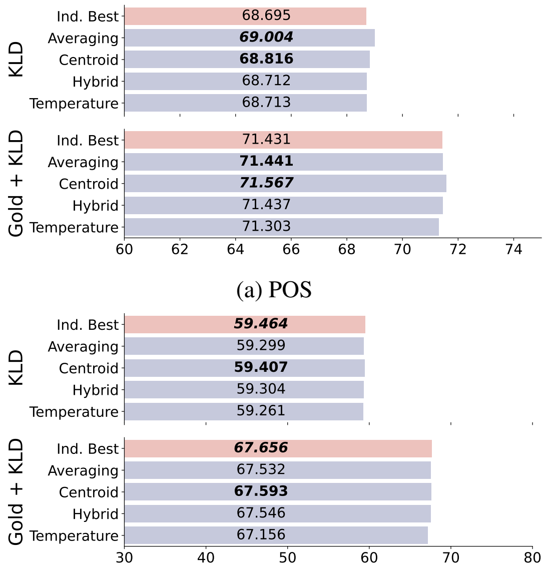
Soft Labels When looking at soft-labels only, the JSC aggregation method provides the most consistent results across tasks, with either the best or second best performance. The hybrid method also offers good uncertainty estimation, especially in the large-data regime of POS tagging and Toxicity detection, though less so for MRE.

Gold + KLD As with the raw performance results, including gold labels in a multi-task setup yields better uncertainty estimation when labeled data is abundant; otherwise using only soft-labels yields better uncertainty estimation.

5.3 Research Questions

RQ1: Best methods for OOD performance. In the out-of-domain setting, we find that among individual soft-labeling techniques, no consistent and clear best performer arises. Aggregating the soft-labels appears to mitigate these fluctuations in performance; in particular, aggregating using the JSC of the individual distributions, which leads to consistently best or near-best performance on all tasks.

RQ2: Does aggregation help? We find that aggregating multiple views of crowd-labels sometimes leads to better performance in the out of distribution setting, but will generally be about as good as the best performing individual methods regardless of poor performance from some individual methods. This is illustrated by the observation that on all tasks in both the multi-task and single-task settings, at least one individual soft labeling method leads to noticeably poorer performance than the best individual method, while aggregation using the JSC is consistently high performing.



(a) POS

(b) Jigsaw

Figure 3: F1 scores for POS and Jigsaw using only the Bayesian methods for aggregation. Best results within each setting are given in ***bold italics***, second best results in **bold**.

RQ3: Uncertainty estimation from soft-labeling. We find that in the absence of hard-labels, different individual soft-labeling methods are inconsistent in their uncertainty estimation across tasks. Again, aggregating these different views of the crowd-sourced labels mitigates these fluctuations. As with raw performance, we find that the JSC is a sensible and consistent choice across tasks in the out-of-distribution setting.

5.4 Analysis

We briefly analyze out results in terms of the relationships between the individual distributions and aggregated distributions. First, we highlight a feature of the JSC, being that its distribution is close to individual distributions which are also close to each other. We do this by correlating two values: the JSD between the JSC aggregated distribution (Q) and an individual distribution (p_m, i.e. $\mathrm{JSD}(Q\|p_m)$), and the average JSD of that distribution to all other individual distributions ($\frac{1}{M-1}\sum_{k!=m}\mathrm{JSD}(p_m\|p_k)$). Doing so yields a statistically significant Pearson correlation of 0.935 ($p \ll 0.05$). This suggests that aggregating using the JSC will lead to distributions closer to the hubs of an ensemble, where many of the individual distributions are similar. This may be desirable if those different views are representative of the problem one is modeling; the downside is the potential to ignore disparate views of the data which could be informative. We leave further exploration of this tradeoff to future work.

Next, we look at differences in the accuracy of the aggregation methods with respect to gold labels in Table 1. We make two notable observations. First, the aggregation methods match or slightly improve the accuracy over the best individual methods, with the exception of the toxicity dataset. Second, for the toxicity dataset, better accuracy with respect to gold labels results in worse performance on the task. This could be explained by the difference in annotators for the labels of the training and test data coupled with the fact that the task is highly subjective.

Finally, we look at performance on the Toxicity and Jigsaw tasks when only using the better performing Bayesian models. These results are given in Figure 3. We find that restricting the distributions to the Bayesian models produces the best performance for POS and closer to the top performing method for Toxicity, suggesting that there is some benefit to selecting good starting distributions for aggregation. While this is difficult to do beforehand without some reliable validation data, it helps to show that aggregation can capture useful training signal from multiple methods while being robust to low-performing individual methods.

6 Conclusion

In this work we present a systematic comparison of soft-labeling techniques from crowd-sourced labels and demonstrate their utility on out-of-domain performance for several text-classification tasks. The out-of-domain setting allows us to observe how learning from crowd-sourced soft-labels enables generalization to unseen domains of data, potentially reflecting the "dark knowledge" imparted by these labels. Given than no consistent best performing model appears, we propose four novel methods for aggregating multiple views of crowd-sourced labels into a combined distribution, demonstrating that doing so leads to consistent performance across tasks despite fluctuations in performance shown by the constituent views. Concretely, we show that using the JSC between the constituent distributions yields high raw performance and good uncertainty estimation. This constitutes a low-cost solution to acquiring reliable soft-labels from crowd-annotations which oftentimes outperform gold labels on out-of-domain data.

Limitations

We propose several methods for learning from multiple-views of crowd annotations; however, acquiring these multiple views requires additional computation for each method which one is aggregating over. While this results in the most consistent performance across tasks and is resilient to low performing individual distributions, better performance is achieved by selecting the best performing individual distributions. However, we do not directly address how to select the best individual distributions. In the same vein, our methods treat all distributions equally, while it may be beneficial to weight each distribution differently. Finally, we look only at NLP tasks and mainly text classification tasks, so we can't say if our results would generalize to other modalities e.g. images.

References

Zeyuan Allen-Zhu and Yuanzhi Li. 2020. Towards Understanding Ensemble, Knowledge Distillation and Self-Distillation in Deep Learning. *CoRR*, abs/2012.09816. 2

Arsenii Ashukha, Alexander Lyzhov, Dmitry Molchanov, and Dmitry P. Vetrov. 2020. Pitfalls of In-Domain Uncertainty Estimation and Ensembling in Deep Learning. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net. 6, 12

Jimmy Ba and Rich Caruana. 2014. Do Deep Nets Really Need to be Deep? In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 2654–2662. 2

Steven Bird. 2006. NLTK: The Natural Language Toolkit. In *ACL 2006, 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, Sydney, Australia, 17-21 July 2006*. The Association for Computer Linguistics. 5, 12

Olivier Bodenreider. 2004. The Unified Medical Language System (UMLS): integrating biomedical terminology. *Nucleic Acids Res.*, 32(Database-Issue):267–270. 11

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 632–642. The Association for Computational Linguistics. 4, 11

Bob Carpenter. 2008. Multilevel Bayesian Models of Categorical Data Annotation. *Unpublished manuscript*, 17(122):45–50. 2, 11

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2005. The PASCAL Recognising Textual Entailment Challenge. In *Machine Learning Challenges, Evaluating Predictive Uncertainty, Visual Object Classification and Recognizing Textual Entailment, First PASCAL Machine Learning Challenges Workshop, MLCW 2005, Southampton, UK, April 11-13, 2005, Revised Selected Papers*, volume 3944 of *Lecture Notes in Computer Science*, pages 177–190. Springer. 4, 11

Alexander Philip Dawid and Allan 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. 2, 3, 11

Anca Dumitrache, Lora Aroyo, and Chris Welty. 2018. Crowdsourcing Semantic Label Propagation in Relation Classification. *CoRR*, abs/1809.00537. 5, 11

Tommaso Fornaciari, Alexandra Uma, Silviu Paun, Barbara Plank, Dirk Hovy, and Massimo Poesio. 2021. Beyond Black & White: Leveraging Annotator Disagreement via Soft-Label Multi-Task Learning. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 2591–2597. Association for Computational Linguistics. 1, 2, 3, 4, 10

Kevin Gimpel, Nathan Schneider, Brendan O'Connor, Dipanjan Das, Daniel Mills, Jacob Eisenstein, Michael Heilman, Dani Yogatama, Jeffrey Flanigan, and Noah A. Smith. 2011. Part-of-Speech Tagging for Twitter: Annotation, Features, and Experiments. In *The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference, 19-24 June, 2011, Portland, Oregon, USA - Short Papers*, pages 42–47. The Association for Computer Linguistics. 5, 7, 11

Mitchell L. Gordon, Kaitlyn Zhou, Kayur Patel, Tatsunori Hashimoto, and Michael S. Bernstein. 2021. The Disagreement Deconvolution: Bringing Machine Learning Performance Metrics In Line With Reality. In *CHI '21: CHI Conference on Human Factors in Computing Systems, Virtual Event / Yokohama, Japan, May 8-13, 2021*, pages 388:1–388:14. ACM. 2

Nitesh Goyal, Ian Kivlichan, Rachel Rosen, and Lucy Vasserman. 2022. Is Your Toxicity My Toxicity? Exploring the Impact of Rater Identity on Toxicity Annotation. *CoRR*, abs/2205.00501. 5, 12

Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. 2015. Distilling the knowledge in a neural network. *CoRR*, abs/1503.02531. 2

Dirk Hovy, Taylor Berg-Kirkpatrick, Ashish Vaswani, and Eduard H. Hovy. 2013. Learning Whom to Trust with MACE. In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 9-14, 2013, Westin Peachtree Plaza Hotel, Atlanta, Georgia, USA*, pages 1120–1130. The Association for Computational Linguistics. 2, 3, 11

Dirk Hovy, Barbara Plank, and Anders Søgaard. 2014. Experiments with crowdsourced re-annotation of a POS tagging data set. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 2: Short Papers*, pages 377–382. The Association for Computer Linguistics. 5, 7, 11

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: Robustly Optimized BERT Pretraining Approach. *CoRR*, abs/1907.11692. 4

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a Large Annotated Corpus of English: The Penn Treebank. *Comput. Linguistics*, 19(2):313–330. 5, 12

Frank Nielsen. 2020. On a Generalization of the Jensen-Shannon Divergence and the Jensen-Shannon Centroid. *Entropy*, 22(2):221. 3, 12

Silviu Paun, Bob Carpenter, Jon Chamberlain, Dirk Hovy, Udo Kruschwitz, and Massimo Poesio. 2018. Comparing Bayesian Models of Annotation. *Trans. Assoc. Comput. Linguistics*, 6:571–585. 2, 11

Joshua C. Peterson, Ruairidh M. Battleday, Thomas L. Griffiths, and Olga Russakovsky. 2019. Human Uncertainty Makes Classification More Robust. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 9616–9625. IEEE. 1, 2, 3, 4, 7, 10

Barbara Plank, Dirk Hovy, and Anders Søgaard. 2014. Linguistically debatable or just plain wrong? In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 2: Short Papers*, pages 507–511. The Association for Computer Linguistics. 5, 11

Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Y. Ng. 2008. Cheap and Fast - But is it Good? Evaluating Non-Expert Annotations for Natural Language Tasks. In *2008 Conference on Empirical Methods in Natural Language Processing, EMNLP 2008, Proceedings of the Conference, 25-27 October 2008, Honolulu, Hawaii, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 254–263. ACL. 4, 11

Alexandra Uma, Tommaso Fornaciari, Dirk Hovy, Silviu Paun, Barbara Plank, and Massimo Poesio. 2020. A Case for Soft Loss Functions. In *Proceedings of the Eighth AAAI Conference on Human Computation and Crowdsourcing, HCOMP 2020, Hilversum, The Netherlands (virtual), October 25-29, 2020*, pages 173–177. AAAI Press. 1, 2, 3, 10

Alexandra Uma, Tommaso Fornaciari, Dirk Hovy, Silviu Paun, Barbara Plank, and Massimo Poesio. 2021. Learning from Disagreement: A Survey. *J. Artif. Intell. Res.*, 72:1385–1470. 1, 2, 3, 4, 5, 6, 11

Chang Wang and James Fan. 2014. Medical Relation Extraction with Manifold Models. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 1: Long Papers*, pages 828–838. The Association for Computer Linguistics. 11

Dustin Wright and Isabelle Augenstein. 2021. Semi-Supervised Exaggeration Detection of Health Science Press Releases. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10824–10836. 5, 11

Alan L. Yuille and Anand Rangarajan. 2001. The Concave-Convex Procedure (CCCP). In *Advances in Neural Information Processing Systems 14 [Neural Information Processing Systems: Natural and Synthetic, NIPS 2001, December 3-8, 2001, Vancouver, British Columbia, Canada]*, pages 1033–1040. MIT Press. 3, 12

A Individual Soft-Labeling Methods

Standard Normalization The standard normalization scheme presented in (Uma et al., 2020) obtains soft-labels for a given sample by transforming a set of crowd-sourced labels into a probability distribution. This is done by normalizing the number of votes given to each label by the total number of annotations for a given sample, as described in Equation 9.

$$p_{stand}(i, y) = \frac{c_{i,y}}{\sum_{\hat{y}} c_{i,\hat{y}}} \tag{9}$$

where $c_{i,y}$ is the number of votes label y received for item i.

Softmax Normalization The standard normalization scheme does not distribute probability mass to any label which receives no votes from any annotator. The works of Peterson et al. (2019); Fornaciari et al. (2021) propose to use the softmax function directly from label vote counts as a way to obtain soft labels for a given sample, as in Equation 10.

$$p_{soft}(i, y) = \frac{e^{c_{i,y}}}{\sum_{\hat{y}} e^{c_{i,\hat{y}}}} \tag{10}$$

This can potentially help to further regularize a model.

Dawid & Skene A common method for aggregating crowd-sourced labels into a single ground-truth label is to treat the true label as a latent variable to be learned from annotations. Several models have been proposed in the literature to accomplish this (Dawid and Skene, 1979; Hovy et al., 2013; Carpenter, 2008), often accounting for other aspects of the annotation problem such as annotator competence and item difficulty. One such method is the Dawid and Skene model (Dawid and Skene, 1979), a highly popular method across fields for aggregating labels from crowd-annotations, which focuses in particular on modeling the true class based on each annotator's ability to correctly identify true instances of a given class. In other words, the model is designed to explain away inconsistencies of individual annotators, which may be desirable for use as a training signal when gold labels are unavailable. To obtain a soft label for a given sample i from this model, we use the posterior distribution of the latent variable c_i which models the true class for a given instance.[2]

MACE Multi-Annotator Competence Estimation (MACE, Hovy et al. 2013)[3] is another Bayesian method popular in NLP which focuses specifically on explaining away poor performing annotators. It does this by learning to differentiate between annotators which likely follow the global labeling strategy of selecting the true underlying label from those which follow a labeling strategy which deviates from this e.g. spamming a single label for every example. To do this, it learns a distribution over the true label for each sample, as well as the likelihood that each annotator is faithfully labeling each sample. For extensive details on both the Dawid and Skene and MACE models, as well as several other Bayesian annotation models, see the survey by Paun et al. (2018).

B Full Dataset Descriptions

Recognizing Textual Entailment (RTE) The first task we consider is recognizing textual entailment (RTE). In the RTE task, a model must predict whether a hypothesis is entailed (i.e. supported) by a given premise. For training, we use the Pascal RTE-1 dataset (Dagan et al., 2005) with crowd-sourced labels from Snow et al. (2008). The dataset consists of 800 premise-hypothesis pairs annotated by 164 different annotators with 10 annotations per pair. The inter-annotator agreement (IAA) is 0.629 (Fleiss κ). As an out-of-domain test set, we use the Stanford Natural Langauge Inference dataset (SNLI) (Bowman et al., 2015), where we transform the task into binary classification by collapsing the "neutral" and "contradiction" classes into a single class.

Medical Relation Extraction (MRE) Medical relation extraction (MRE) seeks to predict what relations hold between different biomedical entities in sentences extracted from biomedical papers. The MRE dataset used for training in this work is the crowd-sourced dataset from (Dumitrache et al., 2018), which collected crowd annotations from 3,984 sentences from PubMed abstracts (Wang and Fan, 2014) annotated by at least 15 annotators for 14 different UMLS (Bodenreider, 2004) relations. Here we focus on the 975 sentence subset which also received expert annotations, specifically for the "cause" relationship. As such, we follow previous work (Uma et al., 2021) and frame the task as a binary classification problem, where a positive label indicates the "cause" relation exists. The IAA for this dataset is 0.857, while the accuracy with respect to the expert gold labels is 76.1%. For testing, we use the causal claim-strength dataset curated from (Wright and Augenstein, 2021), which contains 1,126 sentences from news articles and scientific papers related to health science labeled for causal claim strength (statement of no relation, correlational, conditional causal, and causal). We convert the dataset to a binary classification problem by combining the "conditional causal" and "causal" classes into the positive class and the "correlational" and "no relation" classes into the negative class.

Part-of-Speech Tagging (POS) The POS tagging task is a sequence tagging task, where the goal is to predict the correct part-of-speech for each token in a sentence. For training data, we use the Gimpel dataset from Gimpel et al. (2011) with the crowd-sourced labels provided by Hovy et al. (2014) mapped to the universal POS tag set in Plank et al. (2014). The dataset consists of 1000 tweets (17,503 tokens) labeled with Universal POS tags and annotated by 177 annotators. Each token received at least 5 annotations. The IAA is

[2] Implementation: https://github.com/sukrutrao/Fast-Dawid-Skene

[3] Implementation: https://github.com/dirkhovy/MACE

0.725 and the average annotator accuracy with respect to the gold labels is 67.81%. We use the publicly available sample of the Penn Treebank POS dataset (Marcus et al., 1993) accessed from NLTK (Bird, 2006) as our out-of-domain test set, which consists of 3,914 sentences from Wall Street Journal articles (100,676 tokens).

Toxicity Detection Finally, to measure performance on a highly subjective task, we use the toxicity detection dataset created as a part of the Google Jigsaw unintended bias in toxicity classification competition.[4] The dataset we use comes from Goyal et al. (2022), which annotated 25,500 comments from the original Civil Comments dataset. The pool of annotators is specifically selected and split into multiple rating pools based on self-indicated identity group membership. As this is a highly subjective task, the IAA in terms of Krippendorff's alpha is 0.196. We randomly split the dataset into training and test, and for the test data we use the annotations in the original crowdsourcing task; in other words, using a completely separate annotator pool that isn't selected based on identity groups.

C Evaluation Metrics

F1 We used the sklearn implementation of `precision_recall_fscore_support` for F1 score, which can be found here: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.precision_recall_fscore_support.html. Briefly:

$$p = \frac{tp}{tp + fp}$$

$$r = \frac{tp}{tp + fn}$$

$$F1 = \frac{2 * p * r}{p + r}$$

where tp are true positives, fp are false positives, and fn are false negatives.

Calibrated Log-Likelihood The calibrated log-likelihood is defined in Ashukha et al. (2020) as a method to fairly compare uncertainty estimation between models on the same test set. The key observation is that in order to obtain a fair comparison, one must first perform temperature scaling at the optimal temperature on the classifier output for each model under comparison. Additionally, this temperature must be optimized on an in-domain validation set. The procedure to calculate the calibrated log-likelihood is:

1. Split the **test set** in half, one half for validation and one half for test.
2. Optimize a temperature parameter T to minimize the average negative log-likelihood $-\frac{1}{n}\sum_i \log \tilde{p}(y_i = y_i^*|x_i)$, where $\tilde{p}_i = \text{softmax}(\frac{l_i}{T})$ and l_i is the logits of the classifier, on the validation half of the test set.
3. Measure the temperature scaled log-likelihood on the test half of the test set.

Following the suggestion from Ashukha et al. (2020), we run this procedure 5 times on different splits of the test set and take the average test-half log-likelihood as the result.

D Visualization

Here we plot the JSD between individual methods and the averaging and JSC methods for each dataset in Figure 4.

E CCCP Algorithm for Jensen-Shannon Centroid

Finding the JSC can be done efficiently using methods from convex optimization. In particular, we use the ConCave-Convex procedure (CCCP, Yuille and Rangarajan 2001) developed in Nielsen (2020). The full derivation and definition of the method can be found in Nielsen (2020) Equations 94-104 and Algorithm 1, but at a high level, we can define a categorical distribution p with K classes using the natural parameter θ consisting of $K - 1$ components as:

$$p = \{\theta_{1:(K-1)}, 1 - \sum_{k=1}^{K-1} \theta_k\}$$

The negative entropy of this distribution is then calculated in terms of θ as follows:

$$F(\theta) = -H(\theta) = \sum_{k=1}^{K-1} \theta_k \log \theta_k + (1 - \sum_{k=1}^{K-1} \theta_k) \log(1 - \sum_{k=1}^{K-1} \theta_k) \quad (11)$$

[4] https://www.kaggle.com/competitions/jigsaw-unintended-bias-in-toxicity-classification

which has partial derivatives and inverse gradient:

$$\frac{\partial}{\partial \theta_k} = \log \frac{\theta_k}{1 - \sum_{k=1}^{K-1} \theta_k} \tag{12}$$

$$\theta_k = (\nabla F^{-1}(\eta))_k = \frac{e^{\eta_k}}{1 + \sum_{k=1}^{K-1} e^{\eta_k}} \tag{13}$$

The JSD between two categorical distributions p_1 and p_2 under this view can then be calculated in terms of the negative entropy F defined in Equation 11:

$$\text{JS}(\theta_1 \| \theta_2) = \frac{F(\theta_1) + F(\theta_2)}{2} - F(\frac{\theta_1 + \theta_2}{2})$$

Finally, the hyperparameterless update rule used to find the locally optimum JSC of a set of probability distributions $p_{1:M}$ using their natural parameters $\theta_{1:M}$ is defined in terms of Equation 12 and Equation 13:

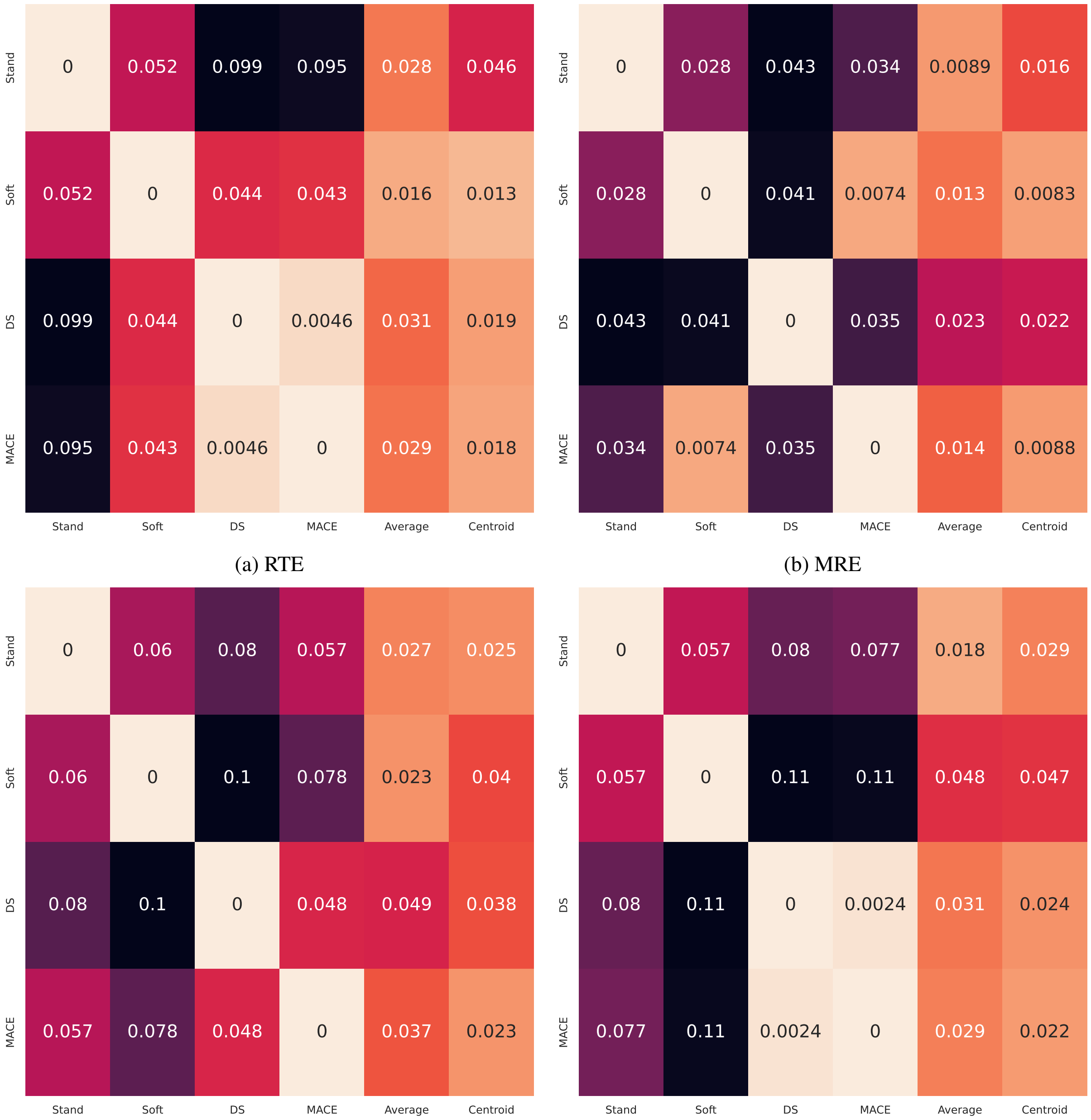
$$\theta^{(t+1)} = (\nabla F)^{-1}(\frac{1}{M} \sum_m F(\frac{\theta_m + \theta^{(t)}}{2})) \tag{14}$$

where $\theta^{(0)} = [f_a(p_{1:M})]_{1:K-1}$.

F Reproducibility

All experiments were run using the RoBERTa base model released in the HuggingFace hub (`roberta-base`[5]) which has 125M parameters. We ran our experiments on a single NVIDIA TITAN RTX with 24GB of RAM. We used a learning rate of 2e-5 with triangular learning rate schedule using 200 warmup steps. POS, RTE, and Toxicity tasks are trained for 5 epochs and MRE is trained for 4 epochs, using the best validation F1 for the final model. The average runtimes are: 50m00s (Toxicity), 1m53s (MRE), 2m28s (POS), 2m39s (RTE).

[5] https://huggingface.co/roberta-base



(a) RTE

(b) MRE

(c) POS

(d) Toxicity

Figure 4: Heatmaps of the average Jensen-Shannon divergence between individual soft labeling methods and average and JS centroid aggregation for (a) RTE, (b) MRE, (c) POS, and (d) Toxicity datasets.