Beyond Black & White: Leveraging Annotator Disagreement via Soft-Label Multi-Task Learning

Tommaso Fornaciari
Università Bocconi
fornaciari@unibocconi.it

Alexandra Uma
Queen Mary University
a.n.uma@qmul.ac.uk

Silviu Paun
Queen Mary University
s.paun@qmul.ac.uk

Barbara Plank
IT University of Copenhagen
bapl@itu.dk

Dirk Hovy
Università Bocconi
dirk.hovy@unibocconi.it

Massimo Poesio
Queen Mary University
m.poesio@qmul.ac.uk

Abstract

Supervised learning assumes that a ground truth label exists. However, the reliability of this ground truth depends on human annotators, who often disagree. Prior work has shown that this disagreement can be helpful in training models. We propose a novel method to incorporate this disagreement as information: in addition to the standard error computation, we use soft labels (i.e., probability distributions over the annotator labels) as an auxiliary task in a multi-task neural network. We measure the divergence between the predictions and the target soft labels with several loss-functions and evaluate the models on various NLP tasks. We find that the soft-label prediction auxiliary task reduces the penalty for errors on ambiguous entities and thereby mitigates overfitting. It significantly improves performance across tasks beyond the standard approach and prior work.

1 Introduction

Usually, the labels used in NLP classification tasks are produced by sets of human annotators. As disagreement between annotators is common, many methods aggregate the different answers into a supposedly correct one (Dawid and Skene, 1979; Carpenter, 2008; Hovy et al., 2013; Raykar et al., 2010; Paun et al., 2018; Ruiz et al., 2019). However, the aggregated labels obtained in this way mask the world's real complexity: instances can be intrinsically ambiguous (Poesio and Artstein, 2005; Zeman, 2010; Plank et al., 2014; Pavlick and Kwiatkowski, 2019), or so challenging to evaluate that considerable disagreement between different annotators is unavoidable. In those cases, it is reasonable to wonder whether the ambiguity is indeed harmful to the models or whether it carries valuable information about the relative difficulty of each instance (Aroyo and Welty, 2015). Several authors followed that intuition, trying ways to incorporate the information about the level of annotator agreement in their models (Sheng et al., 2008; Plank et al., 2014, 2016; Jamison and Gurevych, 2015; Rodrigues and Pereira, 2018; Lalor et al., 2017).

Usually, Deep Learning models compute the error as the divergence between the predicted label distribution and a one-hot encoded gold distribution (i.e., nothing but the gold label has any probability mass). However, for complex tasks, this binary black-and-white notion of truth is not plausible and can lead to overfitting. Instead, we can use a more nuanced notion of truth by comparing against *soft labels*: we collect the probability distributions over the labels given by the annotators, rather than using one-hot encodings with a single correct label. To measure the divergence between probability distributions, we can use well-known measures like the Kullback-Leibler divergence (Kullback and Leibler, 1951), the Jensen-Shannon divergence (Lin, 1991), and the Cross-Entropy, which is also used to quantify the error with one-hot encoded labels. The main impediment to the direct use of soft labels as targets, though, is the lack of universally accepted performance metrics to evaluate the divergence between probability distributions. (Most metrics lack an upper bound, making it difficult to assess prediction quality). Usually, annotations are incorporated into the models without soft labels (Plank et al., 2014; Rodrigues and Pereira, 2018). Where soft labels are used, they are variously filtered according to their distance from the correct labels and then used to weight the training instances rather than as prediction targets. These models still predict only true labels (Jamison and Gurevych, 2015).

In contrast to previous approaches, we use Multi-Task Learning (MTL) to predict a probability distribution over the soft labels as additional output. We jointly model the main task of predicting standard gold labels and the novel auxiliary task of predicting the soft label distributions. Due to the difficulty of interpreting its performance, we do not directly evaluate the distance between the target and the

Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 2591–2597
June 6–11, 2021.

predicted probability distributions. However, the MTL framework allows us to indirectly evaluate its effect on the main task. Exploiting the standard metrics for gold labels, we can also compare the effect of different loss functions for the soft label task. In particular, we propose a standard and an inverse version of the KL-divergence and Cross-Entropy. In previous work (Jamison and Gurevych, 2015), filtering and weighting the training instances according to soft labels did *not* lead to consistent performance improvements. In contrast, we find that the information carried by MTL soft labels *does* significantly improve model performance on several NLP tasks.

Contributions 1) We show that MTL models, trained with soft labels, consistently outperform the corresponding Single-Task Learning (STL) networks, and 2) we evaluate the use of different loss functions for soft labels.

2 MTL with three loss functions

For the experiments, we use different types of neural networks, depending on the type of task. However, we create two versions of each model architecture: an STL model and an MTL model. In STL, we predict the one-hot encoded labels. In MTL, we add the auxiliary task of predicting the soft label distributions to the previous main task.

In both cases, we use Adam optimization (Kingma and Ba, 2014). The loss function for the main task is standard cross-entropy. For the auxiliary task, we have different options. The KL-divergence is a natural choice to measure the difference between the prediction distribution Q and the distribution of soft labels P. However, there are two ways we can do that, depending on what we want to capture.The standard KL-divergence is:

$$D_{KL}(P||Q) = \sum_i P(i)\, log_2 \left(\frac{P(i)}{Q(i)}\right), \quad (1)$$

This measures the divergence from Q to P and encourages a wide Q, because if the model overestimates the regions of small mass from P it will be heavily penalised. The inverse KL-divergence is:

$$D_{KL}(Q||P) = \sum_i Q(i)\, log_2 \left(\frac{Q(i)}{P(i)}\right) \quad (2)$$

This measures the divergence from P to Q and encourages a narrow Q distribution because the model will try to allocate mass to Q in all the places where P has mass; otherwise, it will get a strong penalty.

Considering that we use the auxiliary task to reduce overfitting on the main task, we expect equation 2 to be more effective because it encourages the model to learn a distribution that pays attention to the classes where the annotations possibly agree.

A third option is to directly apply Cross-Entropy. This is actually derived from KL-divergence, the entropy of P added to the KL-divergence:

$$H(P||Q) = H(P) + \sum_i P(i)\, log_2 \left(\frac{P(i)}{Q(i)}\right) \quad (3)$$

$$= \sum_i P(i)\, log_2(Q(i)). \quad (4)$$

Therefore, regular KL-divergence and Cross-Entropy tend to lead to the same performance. For completeness, we report the results of Cross-Entropy as well.

As overall loss of the main and of the auxiliary task, we compute the two's sum. We do not apply any normalization method to the two losses, as unnecessary. We use LogSoftmax activation function for the main task, which is a standard choice for one-hot encoded labels, and standard Softmax for the auxiliary task. Against the distributions of gold (one-hot encoded) and soft labels, both summing up to one, the errors are on the same scale.

We also derive the soft labels using the Softmax function, which prevents the probability of the single labels from falling to zero.

3 Methods

We evaluate our approach on two NLP tasks: POS tagging and morphological stemming. We use the respective data sets from Plank et al. (2014) and Jamison and Gurevych (2015) (where data sets are sufficiently large to train a neural model). In both cases, we use data sets where both one-hot (gold) and probabilistic (soft) labels (i.e., distributions over labels annotations) are available. The code for all models in this paper will be available on github.com/fornaciari.

3.1 POS tagging

Data set For this task, we use the data set released by Gimpel et al. (2010) with the crowd-sourced labels provided by Hovy et al. (2014). The same data set was used by Jamison and Gurevych (2015). Similarly, we use the CONLL Universal

POS tags (Petrov et al., 2012) and 5-fold cross-validation. The soft labels come from the annotation of 177 annotators, with at least five annotations for each instance. Differently from Jamison and Gurevych (2015), however, we also test the model on a completely independent test set, released by Plank et al. (2014). This data set does *not* contain soft labels. However, they are not necessary to *test* our models.

Model We use a tagging model that takes two kinds of input representations, at the character and the word level (Plank et al., 2016). At the character level, we use character embeddings trained on the same data set; at the word level, we use Glove embeddings (Pennington et al., 2014). We feed the word representation into a 'context bi-RNN', selecting the hidden state of the RNN at the target word's position in the sentence. The character representation is then fed into a 'sequence bi-RNN', whose output is its final state. The two outputs are concatenated and passed to an attention mechanism, as proposed by Vaswani et al. (2017). In the STL models, the attention mechanisms' output is passed to a last attention mechanism and to a fully connected layer that gives the output. In the MTL models, the last two components of the STL network (attention + fully connected layer) are duplicated and used for the auxiliary task, providing softmax predictions.

3.2 Morphological stemming

Data set We use the data set used in Jamison and Gurevych (2015), which was originally created by Carpenter et al. (2009). It consists of *(word, stem)*-pairs, and the task is a binary classification task of whether the stem belongs to the word. The soft labels come from 26 unique annotators, and each instance received at least four labels.

Model We represent each *(word, stem)*-pair with the same character embeddings trained for the previous task. Each representation passes to two convolutional/max-pooling layers. We use two convolutional layers with 64 and 128 channels and three windows of 3, 4, and 5 characters size. Their outputs are connected with two independent attention mechanisms (Vaswani et al., 2017). Their output is concatenated and passed directly to the fully connected layers - one for each task -, which provide the prediction. In the MTL models, the concatenation of the attention mechanisms is passed to another fully connected layer, which predicts the soft labels.

4 Experiments and results

4.1 Gold standard and soft labels

To account for the effects of random initializations, we run ten experiments for each experimental condition. During the training, we select the models relying on the F-measure observed on the development set. We report the averaged results for accuracy and F-measure, the metrics used by the studies we compare to. For each task, we compare the STL and MTL models. Where possible, we compare model performance with previous work. Table 1 shows the results. The MTL models significantly outperform the STL ones, and in most cases, the previous state-of-the-art as well. We evaluate STL and MTL's significance via bootstrap sampling, following Berg-Kirkpatrick et al. (2012); Søgaard et al. (2014).

Model	Acc.	F1
POS tag, 5-fold CV		
Jamison and Gurevych	78.9%	-
STL	85.73%	85.00
MTL + KL regular	**86.62%****	**85.90****
MTL + KL inverse	**86.55%****	**85.88****
MTL + Cross-Entropy	**86.76%****	**85.98****
POS tag, separate test set		
Plank et al.	83.6%	-
STL	85.84%	74.56
MTL + KL regular	85.93%	75.04
MTL + KL inverse	**86.29%***	75.04
MTL + Cross-Entropy	**86.27%***	75.13
Stemming		
Jamison and Gurevych	76.6%	-
STL	73.59%	57.57
MTL + KL regular	**75.63%****	55.58
MTL + KL inverse	**77.09%****	**58.41***
MTL + Cross-Entropy	**75.26%****	55.92

Table 1: STL and MTL models with gold and soft labels. Significance: $^{**}: p \leq 0.01$; $^{*}: p \leq 0.05$

4.2 Silver standard and soft labels

Since we did not create the corpora that we use in our experiments, we do not know the details of the gold labels' creation process. However, we verified that the gold labels do not correspond to the classes resulting from the majority voting of the annotations used for the soft labels. Consequently,

the MTL models exploit an additional source of information that is not provided to the STL ones. To validate our hypothesis, we need to exclude that the reason for the MTL's success is not simply that the soft labels inject more information into the models, We ran a set of experiments where the main task was trained on the majority voting (silver) labels from the annotations, rather than on the gold labels. We still performed the tests on the gold labels. In these conditions, both tasks rely on the same source of (imperfect) information, so MTL has no potential advantage over STL. While overall performance drops compared to the results of Table 1, Table 2 shows that the MTL models still maintain a significant advantage over the STL ones. As before, results are averaged over ten independent runs for each condition.

Model	Acc.	F1
POS tagging, 5-fold CV		
STL	75.22%	67.01
MTL + KL regular	**75.82%****	**67.66****
MTL + KL inverse	**76.00%****	**67.76****
MTL + Cross-Entropy	**75.99%****	**67.80****
POS tagging, separate test set		
STL	77.81%	60.59
MTL + KL regular	**78.61%****	**61.68****
MTL + KL inverse	**79.16%****	**61.94****
MTL + Cross-Entropy	**78.49%****	**61.53****
Stemming		
STL	71.34%	58.85
MTL + KL regular	**73.17%****	57.75
MTL + KL inverse	**77.47%****	57.85
MTL + Cross-Entropy	**74.41%****	57.06

Table 2: STL and MTL models with silver and soft labels. Significance: ** : $p \leq 0.01$; * : $p \leq 0.05$

5 Error analysis

To gain further insights about their contributions, we inspect the soft labels' probability distributions, comparing the predictions of STL and MTL models.

We perform the following analysis for the POS and the stemming tasks, and for each kind of loss function in the MTL models. In particular, we consider four-conditions of the predictions: 1) where both STL and MTL gave the correct answer, 2) where both gave the wrong answer, 3) where STL was correct and MTL incorrect, and 4) where MTL was correct and STL incorrect (see confusion matrix in Table 3)

For each of these categories, we compute the relative kurtosis of the soft labels. We choose this measure as it describes how uniform the probability distribution is: whether the annotators agree on a single class, or whether they disperse their votes among different classes.

Not surprisingly, we find the highest average kurtosis where both STL and MTL models give the correct prediction. Both kinds of models find it easier to predict the instances that are also unambiguous for the annotators. The opposite holds as well: the instances where both MTL and STL models are wrong show the lowest mean kurtosis.

More interesting is the outcome where MTL models are correct and STL wrong, and vice-versa. In these cases, the average kurtosis lies between the two previous extremes. Also, we find a consistent trend across data sets and MTL loss-functions: the instances where only the MTL models are correct show a slightly higher kurtosis than those instances where only the STL models give the right answer. To measure the significance of this trend, we apply the Mann-Whitney rank test (Mann and Whitney, 1947). We use a non-parametric test because the kurtoses' distribution is not normal. We find two significant results: when we use Cross-Entropy as MTL loss-function in the POS data set, and with the KL inverse on the Stemming data set. We report the POS results in table 3. Similarly to the previous sections 1 and 2, the results refer to 10 runs of each experimental condition.

This finding suggests that, when dealing with ambiguous cases, the soft labels tend to provide a qualified hint. It is training the models to predict the classes that seem to be the most probable for the annotators.

		MTL	
		correct	incorrect
STL	correct	6.614	5.961
	incorrect	**6.015***	5.727

Table 3: Average soft labels' kurtosis of correctly/incorrectly predicted instances by STL and MTL models (with Cross-Entropy as loss-function) in the POS data set. The kurtosis where only the MTL models are correct is significantly higher than that where only STL models is correct, with * : $p \leq 0.05$

6 Related Work

Several different lines of research use annotation disagreement. One line focuses on the aggregation of multiple annotations before model training. Seminal work includes the proposal by Dawid and Skene (1979), who proposed an Expectation-Maximization (EM) based aggregation model. This model has since influenced a large body of work on annotation aggregation, and modeling annotator competence (Carpenter et al., 2009; Hovy et al., 2013; Raykar et al., 2010; Paun et al., 2018; Ruiz et al., 2019). In our experiments on POS-tagging, we evaluated the possibility of testing Dawid-Skene labels rather than Majority Voting, finding that the performance of the two against the gold standard was mostly the same. Some of these methods also evaluate the annotators' expertise (Dawid and Skene, 1979; Raykar et al., 2010; Hovy et al., 2013; Ruiz et al., 2019). Others just penalize disagreement (Pan et al., 2019). The second line of work focuses on filtering out presumably low quality data to train on the remaining data (Beigman Klebanov and Beigman, 2014; Jamison and Gurevych, 2015). However, such filtering strategies require an effective filtering threshold, which is non-trivial; relying only on high-agreement cases also results in worse performance (Jamison and Gurevych, 2015). Some studies (Goldberger and Ben-Reuven, 2016; Han et al., 2018b,a) treat disagreement as a corruption of a theoretical gold standard. Since the robustness of machine learning models is affected by the data annotation quality, reducing noisy labels generally improves the models' performance. The closest to our work are the studies of Cohn and Specia (2013) and Rodrigues and Pereira (2018), who both use MTL. In contrast to our approach, though, each of their tasks represents an annotator. We instead propose to learn from both the gold labels *and* the distribution over multiple annotators, which we treat as soft label distributions in a single auxiliary task. Compared to treating each annotator as a task, our approach has the advantage that it requires fewer output nodes, which reduces the number of parameters. To our knowledge, the only study that directly uses soft labels is the one by Lalor et al. (2017). Different from our study, they assume that soft labels are available only for a subset of the data. Therefore they use them to fine-tune STL networks. Despite this methodological difference, their findings support this paper's intuition that soft labels carry signal rather than noise.

In a broad sense, our study belongs to the research area of regularization methods for neural networks. Among them, label smoothing (Pereyra et al., 2017) penalizes the cases of over-confident network predictions. Both label smoothing and soft labels reduce overfitting regulating the loss size. However, label smoothing relies on the gold labels' distribution, not accounting for the instances' inherent ambiguity, while soft labels selectively train the models to reduce the confidence when dealing with unclear cases, not affecting the prediction of clear cases. Disagreement also relates to the issue of annotator biases (Shah et al., 2020; Sap et al., 2019; Hovy and Yang, 2021), and our method can provide a possible way to address it.

7 Conclusion

We propose a new method for leveraging instance ambiguity, as expressed by the probability distribution over label annotations. We set up MTL models to predict this label distribution as an auxiliary task in addition to the standard classification task. This setup allows us to incorporate uncertainty about the instances' class membership into the model. Across two NLP tasks, three data sets, and three loss functions, we always find that our method significantly improves over the STL performance. While the performance difference between the loss functions is not significant, we find that the inverse version of KL gives the best results in all the experimental conditions but one. This finding supports our idea of emphasizing the coders' disagreement during training. We conjecture that predicting the soft labels acts as a regularizer, reducing overfitting. That effect is especially likely for ambiguous instances, where annotators' label distributions differ especially strongly from one-hot encoded gold labels.

Acknowledgements

DH and TF are members of the Data and Marketing Insights Unit at the Bocconi Institute for Data Science and Analysis. AU, MP and SP were funded by the DALI project, ERC Grant 695662. We would like to thank the various reviewers who suggested valuable edits and additions.

References

Lora Aroyo and Chris Welty. 2015. Truth is a lie: Crowd truth and the seven myths of human annotation. *AI Magazine*, 36(1):15–24.

Beata Beigman Klebanov and Eyal Beigman. 2014. Difficult cases: From data to learning, and back. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 390–396, Baltimore, Maryland. Association for Computational Linguistics.

Taylor Berg-Kirkpatrick, David Burkett, and Dan Klein. 2012. An empirical investigation of statistical significance in NLP. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 995–1005, Jeju Island, Korea. Association for Computational Linguistics.

Bob Carpenter. 2008. Multilevel bayesian models of categorical data annotation. Available as http://lingpipe.files.wordpress.com/2008/11/carp-bayesian-multilevel-annotation.pdf.

Bob Carpenter, Emily Jamison, and Breck Baldwin. 2009. Building a stemming corpus: Coding standards. https://lingpipe-blog.com/2009/02/25/.

Trevor Cohn and Lucia Specia. 2013. Modelling annotator bias with multi-task Gaussian processes: An application to machine translation quality estimation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 32–42, Sofia, Bulgaria. Association for Computational Linguistics.

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

Kevin Gimpel, Nathan Schneider, Brendan O'Connor, Dipanjan Das, Daniel Mills, Jacob Eisenstein, Michael Heilman, Dani Yogatama, Jeffrey Flanigan, and Noah A Smith. 2010. Part-of-speech tagging for twitter: Annotation, features, and experiments. Technical report, Carnegie-Mellon Univ Pittsburgh Pa School of Computer Science.

Jacob Goldberger and Ehud Ben-Reuven. 2016. Training deep neural-networks using a noise adaptation layer. *ICLR 2017*.

Bo Han, Jiangchao Yao, Gang Niu, Mingyuan Zhou, Ivor Tsang, Ya Zhang, and Masashi Sugiyama. 2018a. Masking: A new perspective of noisy supervision. In *Advances in Neural Information Processing Systems*, pages 5836–5846.

Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi Sugiyama. 2018b. Co-teaching: Robust training of deep neural networks with extremely noisy labels. In *Advances in neural information processing systems*, pages 8527–8537.

Dirk Hovy, Taylor Berg-Kirkpatrick, Ashish Vaswani, and Eduard Hovy. 2013. Learning whom to trust with MACE. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1120–1130, Atlanta, Georgia. Association for Computational Linguistics.

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 (Volume 2: Short Papers)*, pages 377–382, Baltimore, Maryland. Association for Computational Linguistics.

Dirk Hovy and Diyi Yang. 2021. The importance of modeling social factors of language: Theory and practice. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics*, Minneapolis, Minnesota. Association for Computational Linguistics.

Emily Jamison and Iryna Gurevych. 2015. Noise or additional information? leveraging crowdsource annotation item agreement for natural language tasks. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 291–297.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

S. Kullback and R. A. Leibler. 1951. On information and sufficiency. *Ann. Math. Statist.*, 22(1):79–86.

John P Lalor, Hao Wu, and Hong Yu. 2017. Soft label memorization-generalization for natural language inference. *arXiv preprint arXiv:1702.08563*.

Jianhua Lin. 1991. Divergence measures based on the shannon entropy. *IEEE Transactions on Information theory*, 37(1):145–151.

Henry B Mann and Donald R Whitney. 1947. On a test of whether one of two random variables is stochastically larger than the other. *The annals of mathematical statistics*, pages 50–60.

Boyuan Pan, Yazheng Yang, Zhou Zhao, Yueting Zhuang, Deng Cai, and Xiaofei He. 2019. Discourse marker augmented network with reinforcement learning for natural language inference. *arXiv preprint arXiv:1907.09692*.

Silviu Paun, Bob Carpenter, Jon Chamberlain, Dirk Hovy, Udo Kruschwitz, and Massimo Poesio. 2018. Comparing Bayesian models of annotation. *Transactions of the Association for Computational Linguistics*, 6:571–585.

Ellie Pavlick and Tom Kwiatkowski. 2019. Inherent disagreements in human textual inferences. *Transactions of the Association for Computational Linguistics*, 7:677–694.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Gabriel Pereyra, George Tucker, Jan Chorowski, Łukasz Kaiser, and Geoffrey Hinton. 2017. Regularizing neural networks by penalizing confident output distributions. *arXiv preprint arXiv:1701.06548*.

Slav Petrov, Dipanjan Das, and Ryan McDonald. 2012. A universal part-of-speech tagset. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 2089–2096, Istanbul, Turkey. European Language Resources Association (ELRA).

Barbara Plank, Dirk Hovy, and Anders Søgaard. 2014. Learning part-of-speech taggers with inter-annotator agreement loss. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 742–751, Gothenburg, Sweden. Association for Computational Linguistics.

Barbara Plank, Anders Søgaard, and Yoav Goldberg. 2016. Multilingual part-of-speech tagging with bidirectional long short-term memory models and auxiliary loss. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 412–418, Berlin, Germany. Association for Computational Linguistics.

Massimo Poesio and Ron Artstein. 2005. The reliability of anaphoric annotation, reconsidered: Taking ambiguity into account. In *Proc. of ACL Workshop on Frontiers in Corpus Annotation*, pages 76–83.

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(Apr):1297–1322.

Filipe Rodrigues and Francisco C Pereira. 2018. Deep learning from crowds. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Pablo Ruiz, Pablo Morales-Álvarez, Rafael Molina, and Aggelos K Katsaggelos. 2019. Learning from crowds with variational gaussian processes. *Pattern Recognition*, 88:298–311.

Maarten Sap, Dallas Card, Saadia Gabriel, Yejin Choi, and Noah A. Smith. 2019. The risk of racial bias in hate speech detection. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1668–1678, Florence, Italy. Association for Computational Linguistics.

Deven Santosh Shah, H. Andrew Schwartz, and Dirk Hovy. 2020. Predictive biases in natural language processing models: A conceptual framework and overview. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5248–5264, Online. Association for Computational Linguistics.

Victor S Sheng, Foster Provost, and Panagiotis G Ipeirotis. 2008. Get another label? improving data quality and data mining using multiple, noisy labelers. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 614–622. ACM.

Anders Søgaard, Anders Johannsen, Barbara Plank, Dirk Hovy, and Hector Martínez Alonso. 2014. What's in a p-value in NLP? In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning*, pages 1–10, Ann Arbor, Michigan. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Daniel Zeman. 2010. Hard problems of tagset conversion. In *Proceedings of the Second International Conference on Global Interoperability for Language Resources*, pages 181–185.