# Making the Most of Crowd Information: Learning and Evaluation in AI tasks with Disagreements

Alexandra Nnemamaka Uma

PhD thesis

School of Electronic Engineering and Computer Science
Queen Mary University of London

August 2021

## Abstract

There is plenty of evidence that humans disagree on the interpretation of many tasks in Natural Language Processing (NLP) and Computer Vision (CV), from objective tasks rooted in linguistics such as part-of-speech tagging to more subjective (observer-dependent) tasks such as classifying an image or deciding whether a proposition follows from a certain premise. While most learning in Artificial Intelligence (AI) still relies on the assumption that a single interpretation, captured by the gold label, exists for each item, a growing research body in recent years has focused on learning methods that do not rely on this assumption. Rather, they aim to learn ranges of truth amidst disagreement. This PhD research makes a contribution to this field of study.

Firstly, we analytically review the evidence for disagreement on NLP and CV tasks, focusing on tasks where substantial datasets with such information have been created. As part of this review, we also discuss the most popular approaches to training models from datasets containing multiple judgments and group these methods together according to their handling of disagreement. Secondly, we make three proposals for learning with disagreement; soft-loss, multi-task learning from gold and crowds, and automatic temperature-scaled soft-loss. Thirdly, we address one gap in this field of study – the prevalence of hard metrics for model evaluation even when the gold assumption is shown to be an idealization – by proposing several previously existing metrics and novel soft metrics that do not make this assumption and analyzing the merits and assumptions of all the metrics, hard and soft. Finally, we carry out a systematic investigation of the key proposals in learning with disagreement by training them across several tasks, considering several ways to evaluate the resulting models and assessing the conditions under which each approach is effective. This is a key contribution of this research as research in learning with disagreement do not often test proposals across tasks, compare proposals with a variety of approaches, or evaluate using both soft metrics and hard metrics.

The results obtained suggest, first of all, that it is essential to reach a consensus on how to evaluate models. This is because the relative performance of the various training methods is critically affected by the chosen form of evaluation. Secondly, we observed a strong dataset effect. With substantial datasets, providing many judgments by high-quality coders for each item, training directly with soft labels achieved better results than training from aggregated or even gold labels. This result holds for both hard and soft evaluation. But when the above conditions do not hold, leveraging both gold and soft labels generally achieved the best results in the hard evaluation. All datasets and models employed in this paper are freely available as supplementary materials.

# Acknowledgements

I would like to thank everyone was a part of this incredible journey. Firstly, I would like to thank Massimo Poesio, my supervisor and mentor for the past four years. Thank you for taking a chance and for your steadfast guidance, coaching and encouragement throughout this process; it has been invaluable. I would also like to thank the other members of my advisory board, Prof. Matthew Purver, and Prof. Ioannis Patras whose reviews, comments and advice have helped shaped this work.

Secondly, I would like to thank the members of the Cognitive Science Research group and Computer Science department here at Queen Mary for their help with everything from locating a particular building, to suggesting a well needed lunch break, to help with some new technology. I would also thank the members of the DALI research group and the soft labelling group especially Juntao Yu, Silviu Paun, Tommaso Fornaciari, Barbara Plank and Dirk Hovy, who I've worked with directly for the past few years; it has been wonderful learning from and working with you.

To my personal support system, my mum, dad, siblings, friends and friends who have become family, I want to say a big thank you. Thank you for putting up with my incessant 'PhD and AI talk'. Thank you for being there to encourage me when I didn't have the strength to pull myself up. Thank you for your prayers and support; I wouldn't be here without it. I am grateful to have you in my life. Thank you everyone. This PhD would not be what it is without all of you.

This research was supported by the DALI project, ERC Grant 695662.

# Declaration

I, Alexandra Nnemamaka Uma, confirm that the research included within this thesis is my own work, or that where it has been carried out in collaboration with, or supported by others, this is duly acknowledged. The collaborators are listed in the associated publications at the end of Chapter 1.

I attest that I have exercised reasonable care to ensure that the work is original, and does not to the best of my knowledge break any UK law, infringe any third party's copyright or other Intellectual Property Right, or contain any confidential material.

I accept that the College has the right to use plagiarism detection software to check the electronic version of the thesis.

I confirm that this thesis has not been previously submitted for the award of a degree by this or any other university.

Alexandra Nnemamaka Uma
August 2021

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

## 1.1 Motivation

Modern research in Cognitive Science and Artificial Intelligence (AI) is driven by the availability of large datasets annotated with human judgments [Ide and Pustejovsky, 2017]. These data instances and their corresponding labels are not only used to train and test computational models, but also to provide data-driven evidence of linguistic phenomena, complementing a linguist's intuition. In addition, they can be used to compute statistics about the frequencies of certain phenomena [de Marneffe and Potts, 2017].

The simplest way to create an annotated dataset is to appoint a single **expert**, proficient at the task and motivated either by altruism or a financial incentive, to provide the labels for all the data instances (or items). This person is often a project member or a (trained) hired student. This approach, however, is only feasible for the small-to-medium scale annotations that were the norm until ten years ago but not for the much larger datasets required now. In addition, the quality of the data produced this way is overly dependent on the level of expertise of this sole annotator and their skillfulness at annotation. Furthermore, the data is implicitly encoded with any bias the annotator may have about the subject matter. To mitigate these limitations of sole annotation, the approach adopted in most large-scale annotation projects is for several experts to carry out the annotations. Typically, 2-3 annotations for each item are provided by a few experts, and a final adjudication step is carried out to produce a single label for each item called the **gold label**. This was the strategy used to annotated the best known NLP corpora ONTONOTES [Pradhan et al., 2011, Hovy et al., 2006]. However, using experts is very expensive, prohibitively so for large-scale annotation projects. Thus, a third alternative has gained increasing popularity: to source the annotations from a "crowd" of people, typically (but not always) non-experts. This approach is called **crowdsourcing** [Snow et al., 2008, Poesio et al., 2017]. This crowd can be recruited by offering small financial pay-outs (in which case the approach is sometimes known as **microtask crowdsourcing**), or by redressing the task as a game (so-called **game-with-a-purpose**) that people are willing to play for fun without being coerced to do so [von Ahn and Dabbish, 2008]. Using crowdsourcing, the annotations can be collected faster and at the fraction of the cost that it takes to collect them from an expert or a few experts.

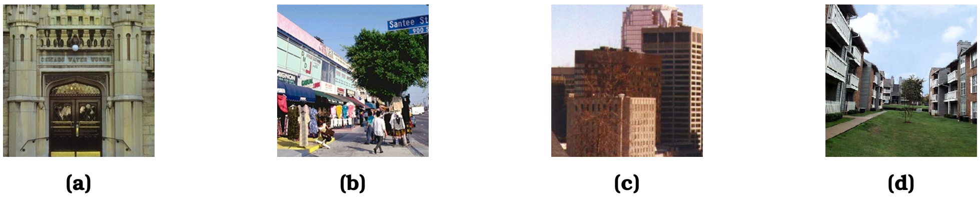
Notwithstanding these differences, most annotation projects assume that a single preferred interpretation (an objective truth) exists for each instance to be annotated; and that where available, the gold label captures this objective truth. But research has shown this **gold assumption** to be an idealization at best, both in natural language processing and computer vision. Every large-scale annotation project frequently encounters cases on which humans **disagree**. In some cases, these disagreements are due to misunderstandings or problems with the annotation interface. In other cases, they are due to poorly specified annotation schemes, or a result of the difficulty of the task, which causes annotators to stumble. In yet more cases, disagreements arise because the interpretation is inherently ambiguous or unclear. For example, for anaphoric / coreference annotation, Poesio et al. [2007] discussed **justified sloppiness** in anaphoric reference, illustrated in example (1.1).

```
(1.1) 3.1     M: can we .. kindly hook up
      3.2      : uh
      3.3      : engine E2 to the boxcar at ..
                Elmira
      4.1     S: ok
      5.1     M: +and+ send it to Corning
      5.2      : as soon as possible please
      6.1     S: okay
               [2sec]
      7.1     M: do let me know when it gets
                there
      8.1     S: okay it'll /
      8.2      : it should get there at 2 AM
      9.1     M: great
      9.2      : uh can you give the
      9.3      : manager at Corning instructions
                that
      9.4      : as soon as it arrives
      9.5      : it should be filled with
                oranges
      10.1    S: okay
      10.2     : then we can get that filled
```

In this example, it is not clear whether the pronoun *it* in 5.1 (in blue) refers to *the engine E2* which has been hooked up to *the boxcar at Elmira*, to the boxcar itself, or indeed whether that matters. It's only at utterance 9.5 that we get evidence that *it* probably referred to *the boxcar at Elmira*, since it is only boxcars that can be filled with oranges. Evidence that subjects disagree on such cases was discussed, e.g., in Poesio and Artstein [2005] and Poesio et al. [2006], and similar cases of disagreements on anaphoric labels have been found in all large-scale anaphoric annotation projects [Versley, 2008, Recasens et al., 2011, Pradhan et al., 2012].

Indeed, disagreements are frequent in all areas of NLP and in all large-scale annotation projects. The NLP community has realized from the start that it makes no sense to consider gold labels/targets as objective truth in applications such as machine translation, summarization, and natural language generation, where human creativity plays a role and has developed specialized training and evaluation methods for such applications. Recently, the field has tackled classification tasks that involve labelling text according to inherently subjective judgments, such as sentiment analysis [Kenyon-Dean et al., 2018] or offensive language detection [Basile, 2020]. It would be clearly misguided to rely on gold labels for training or evaluation in such tasks, as doing so would set one subjective interpretation over all alternatives. Disagreements in interpretation have also been found in annotation projects, such as natural language inference, that ask annotators to make complex judgements [Pavlick and Kwiatkowski, 2019]. But disagreements in interpretation are not limited to these complex cases; in fact, they are commonly found even in annotation projects concerned with what might have been thought of as objective and “simple” aspects of language, from part-of-speech tagging [Plank et al., 2014b] to wordsenses [Passonneau et al., 2012] and semantic role labelling [Dumitrache et al., 2019]: These aspects of language are what this thesis focuses on.

In computer vision as well, the assumption that an objective true class exists for all items to be classified has proven an idealization. In many widely used crowdsourced datasets for computer vision, different coders assign equally plausible labels to the same items. Consider for instance the task of object identification in images. Examples (a), (b), and (c) in Figure 1.1, discussed in [Rodrigues et al., 2017], are from the LabelMe dataset [Russell et al., 2008]. Due to the overlap between the labels, the judgments of coders are highly subjective. The gold label for (a) is ‘inside city’, and one annotator chose that label as well, but two other annotators chose ‘tall building’. The gold label for (b) is ‘street’, and again, this was produced by one of the annotators, but two others chose ‘inside city’. The same is true for (c). For (d), none of the annotators chose the gold, ’street’; all chose ’inside city’. Clearly, in all of these cases, annotator labels can be considered acceptable even if they differ from the gold. In these and similar cases, the gold label though treated like an objective truth is shown to be the bias of the expert or an arbitrary choice among plausible alternatives.



(a) (b) (c) (d)

**Figure 1.1:** Examples from the LabelMe dataset [Russell et al., 2008]

Possibly the most widely adopted approach to dealing with disagreements in crowdsourced data is a source-filter model, i.e., persist in the assumption that there exists a single objective truth that is merely obfuscated by the disagreements, and use an **aggregation method** over the noisy annotations to find the true label, a latent parameter [Dawid and Skene, 1979, Carpenter, 2008, Whitehill et al., 2009, Hovy et al., 2013, Passonneau and Carpenter, 2013]. Another approach, also based on the idealization of gold labels is to **filter** away items with substantial disagreements; exclude them from the training set or at the very least from the evaluation set [Beigman-Klebanov and Beigman, 2009]). Some researchers though, have that disagreement is signal, not noise to be filtered out or evened out by aggregation [Aroyo and Welty, 2015, Jamison and Gurevych, 2015, Sharmanska et al., 2016, Plank et al., 2014a]–i.e., that disagreements provide information that is useful for learning. And, various models have been proposed to leverage all of the information provided by annotators, including information about disagreements [Plank et al., 2014a, Aroyo and Welty, 2015, Jamison and Gurevych, 2015, Sharmanska et al., 2016]; some models do not rely on gold labels at all [Sheng et al., 2008, Sharmanska et al., 2016, Guan et al., 2018, Rodrigues and Pereira, 2018, Firman et al., 2018, Peterson et al., 2019, Uma et al., 2020].

Most research in *learning to classify from crowds* involve proposing a novel method for learning from crowds amidst disagreements. Usually, the proposed method is targeted at a specific task (exemplified by a crowdsourced dataset) and shown to be successful in learning that task from crowds – usually (1) the scope of the method is limited to this dataset alone and the ability of the method to generalize across tasks is not tested [Plank et al., 2014a, Albarqouni et al., 2016, Dumitrache et al., 2018a, Pavlick and Kwiatkowski, 2019]; (2) often there is little investigation into how the success of the method is impacted by the nature of the task and the levels and sources of disagreement in the dataset [Rodrigues and Pereira, 2018]; and (3) in nearly all the proposals, model success is measured using hard metrics like accuracy or f1 in relation to gold labels. This PhD seeks to address these limitations. In addition to proposing new ways of learning with disagreement, this work comprehensively surveys the evidence for disagreements on judgments required from AI systems and the range of approaches that have emerged in computational linguistics and in computer vision. Beyond reviewing them, this research categorizes these methods based on their approach to handling disagreement, and compares them with each other on some of the key datasets providing evidence about disagreement. In this thesis, the discussion is limited to datasets for tasks usually considered objective. The conditions under which these methods are effective are assessed using not only hard evaluation metrics, but soft evaluation metrics are proposed and used - this is a key contribution of this work

## 1.2 Aim and Thesis Structure

Thus, the aim of this PhD is to study the evidence for disagreements on annotation judgements and the range of approaches to learning from such disagreements that have emerged in computer vision and computational linguistics systems, and, having systematically and comprehensively undertaken such a study, develop state-of-the art disagreement-aware models. Importantly, this work also proposes metrics for evaluating these models without assuming a single correct label per item. To advance this aim, this PhD addresses several research questions:

- ***RQ1***: *What is the evidence for the presence of label uncertainty and ambiguity in NLP and Computer Vision and is disagreement in crowd annotations evidence of ambiguity?* **chapter 2** summarizes arguments and theories highlighting uncertainty, vagueness and ambiguity in classification. It also summarizes the analysis of crowd-annotated datasets carried out by several key researchers on the incidence of disagreement and its relationship with uncertainty and ambiguity. This chapter also contains an overview of the methods for learning from crowd annotations and groups them into approaches based on their philosophy of noise, label uncertainty and ambiguity.

- ***RQ2***: *What is the most appropriate way of evaluating a model on datasets which provide the whole range of crowd opinions, if we don't assume that every item can be interpreted in a single way?* **Chapter 3** contains a case for soft evaluation. The metrics explored in this Chapter are used throughout this work, providing insights to **RQ2**. Chapter 6 analyzes the results of the various evaluation metrics across methods and datasets.

- ***RQ3***: *Can models trained using multiple annotations/interpretations, without assuming gold labels, achieve similar or better performance as methods that rely on gold labels alone?* **Chapter 4** makes the case for a *soft loss* method of training models using crowd labels. It provides further evidence to the findings of Peterson et al. [2019], that training using crowd labels can outperform training using gold labels under certain conditions.

- ***RQ4***: ***(a)*** *Can information from crowd annotations be used in conjunction with gold labels to build better models compared to learning from gold labels only?* ***(b)*** *In case the answer to (a) is positive, what is the best way of leveraging crowd information in addition to gold labels?* **Chapter 5** contains two proposals for augmenting gold training with crowd information using the multi-task learning paradigm.

- ***RQ5***: *Among the approaches for learning from crowds, is there an absolute best method for every task?* **Chapter 6** provides a detailed analysis and experimental survey of the methods for learning from crowds, comparing the methods with each other and with learning from gold labels.

To further this line of inquiry in the wider research community, we propose a shared task at the "The 15th International Workshop on Semantic Evaluation". The proceedings from this shared task were published in Uma et al. [2021a] and are included as **Chapter 7** of this work.

## 1.3 Published Work

A lot of the material in this thesis has been published in national and international publications. While the copyright to these published works belong to the various publishers (copyright quoted below), I, as author of these publications, retained the right to use the material in future works of my own authorship such as this thesis. The chapters and their associated publications are listed below (journal publications are highlighted in bold):

- Sections of Chapters 1 to 6 were written up as a journal paper to appear in **Journal of Artificial Intelligence Research (JAIR)** [Uma et al., 2021b]. Copyright © 2020, AI Access Foundation.
- A shorter version of Chapter 4 was presented at the AAAI Conference on Human Computation and Crowdsourcing (HCOMP) [Uma et al., 2020]. Copyright © 2020, Association for the Advancement of Artificial Intelligence.
- Sections of Chapter 5 was published in the Proceedings of the North American Chapter of the Association for Computational Linguistics (NAACL) [Fornaciari et al., 2021]. Copyright © 1963–2021 ACL.
- Chapter 7 was published in The 15th International Workshop on Semantic Evaluation at ACL-IJCNLP 2021 [Uma et al., 2021a]. Copyright © 1963–2021 ACL.
- An extended version of Chapter 8 was submitted to the **Human-Centered AI: Crowd Computing topic of the Frontiers Journal**. Copyright © 2007-2021 Frontiers Media SA (pending).

**Other Publications**

- “Anaphora Resolution with the ARRAU Corpus" [Poesio et al., 2018]
- “A Crowdsourced Corpus of Multiple Judgments and Disagreement on Anaphoric Interpretation" [Poesio et al., 2019]
- “A Cluster Ranking Model for Full Anaphora Resolution" [Yu et al., 2020]
- “We Need to Consider Disagreement in Evaluation"[Basile et al., 2021]

# Chapter 2

# Background

*In the introductory chapter, we motivated the problem of learning from crowdsourced data containing disagreement. In this Chapter, we study the evidence of and approaches to learning from disagreement. We highlight various crowdsourced datasets containing disagreement and analyse the data for evidence of disagreement as useful information about the nature of the task. We also provide systematic review overview of methods for learning from crowds, categorizing them according to their approach to dealing with disagreement.*

## 2.1 Overview

In this Chapter, we study the evidence of and approaches to learning from disagreement. Section 2.2 reviews several tasks and datasets that provide evidence for disagreement in human interpretation. In doing this we provide answers to **RQ1**: What is the evidence for the presence of label uncertainty and ambiguity in NLP and Computer Vision and is disagreement in crowd annotations evidence of ambiguity? Section 2.3 describes the patterns and sources of disagreement, Finally, Section 2.4 contains and in-depth analysis of the the methods of learning from crowd annotations in literature, grouping them in to approaches based on how they handle disagreement.

## 2.2 Disagreements in NLP and Computer Vision: evidence and resources

As already mentioned, there is extensive literature demonstrating the extent to which humans disagree on many aspects of interpretation in NLP and CV. In this Section, we review some of this evidence, focusing on work that also created datasets that preserved these disagreements. Our discussion is thus structured around NLP and CV research on learning from disagreements: This thesis limits the discussion on learning with disagreement to tasks for which the gold assumption is usually held (i.e. tasks assumed to be objective rather than subjective tasks such as offensive language detection and opinion classification). The datasets analyzed here form the basis of the studies written up in the rest of this thesis.

We chose tasks with the aim of using deep learning methods that reflect the current state of the art to improve upon previous experimental studies on learning from disagreement, in particular Jamison and Gurevych [2015]. This constraint restricted us to tasks for which datasets large enough to train such models exist; our rule of thumb was to consider only datasets with at least 1,000 items. The one exception was the area of Recognizing Textual Entailment (RTE) / Natural Language Inference, which includes some vital work on disagreements in interpretation [Pavlick and Kwiatkowski, 2019], as well as the dataset by [Snow et al., 2008]. This dataset consists of only 800 items, but was included due to its ubiquitous use in research on crowdsourcing and aggregation [1].

The NLP tasks we selected include Part-of-Speech (POS) tagging, which originated the Gimpel corpus [Plank et al., 2014b,a, Jamison and Gurevych, 2015]); Information status (IS) classification, a simplified version of the anaphoric interpretation task studied in some of the early work on disagreements [Poesio and Artstein, 2005, Poesio et al., 2006] and for which we could leverage the largest NLP corpus providing multiply annotated data, *Phrase Detectives* [Poesio et al., 2019]; (Medical) Relation Extraction (MRE), extensively studied in the CrowdTruth project [Aroyo and Welty, 2015, Dumitrache, 2019, Dumitrache et al., 2019] which resulted in the creation of several datasets including the dataset used in this study [Dumitrache et al., 2018a]; and Recognizing Textual Entailment (RTE), which led to the development of the Snow *et al.* corpus used, e.g., in [Snow et al., 2008, Jamison and Gurevych, 2015]. Among the CV tasks, we studied Image Classification (IC), in which two important datasets originated for the study of learning from disagreement: the LabelMe corpus (IC-LABELME), a crowdsourced version which was created by Rodrigues and Pereira [2018], and the CIFAR-10H corpus recently crowdsourced by Peterson et al. [2019]. This Section briefly discusses each of these tasks, their respective datasets, and the evidence about disagreement resulting from them. We use a standardized format for the task descriptions to facilitate comparison, and summarize the characteristics of the datasets in Section 2.2.6.

### 2.2.1 Part-of-Speech Tagging

POS tagging is the task of assigning Part-Of-Speech tags such as noun or verb to every word in a text. It is thought of as reflecting a very basic aspect of human lexical / syntactic competence, and as such, little or no disagreement is expected on the judgments of coders asked to carry out this type of annotation. Contrary to this expectation, one of the best-known studies in the area of learning from disagreements was motivated by the observation that annotators systematically disagreed even on such a supposedly simple linguistic task such as part-of-speech tagging [Plank et al., 2014b].[2]

[1] the MRE dataset falls slightly short of the mark with 975 examples but was central to the Crowd Truth Project, a leading research of learning with disagreement as noise Dumitrache et al. [2018b]

[2] See also [Manning, 2011] for an earlier discussion of the same issue.

Plank *et al.* found systematic disagreements between, e.g., adpositions (ADP) and particles (PRT) as in *get out*; adjectives (ADJ) and nouns, as in *stone lion*; and adjectives and adverbs (ADV), e.g., in *see you later*. They found the same disagreements between experts and non-experts, and across text types. Plank *et al.* investigated the nature of these disagreements, finding that while some disagreements are a result of annotation error, others are evidence that the category of certain items is linguistically debatable. They further discovered that making the annotation guidelines increasingly more detailed did not eliminate these linguistically debatable disagreements or "hard cases" [Plank et al., 2014b]. They thus hypothesized that these disagreements are a result of label uncertainty and can be used to inform the learning process.

**The dataset** The analysis by Plank et al. [2014b] was carried out as part of the creation of one of the best known dataset for research on learning from disagreement, and the first dataset chosen for this study. This dataset–henceforth, abbreviated as GIMPEL-POS–builds upon the [Gimpel et al., 2011] corpus of POS labels for Twitter posts. Plank et al. [2014b] mapped the Gimpel tags to the universal tag set [Petrov et al., 2012], using these tags as gold, and collected at least 5 crowdsourced label per token from 177 annotators. The dataset consists of over 14 thousand examples, and was already used in [Plank et al., 2014a, Jamison and Gurevych, 2015].

**Annotations and annotators** The size of the crowd employed to collect judgments is essential to ensure sufficient quality for the crowdsourced labels [Snow et al., 2008]. Additionally, a number of studies [Poesio and Artstein, 2005, Dumitrache, 2019, Peterson et al., 2019] have demonstrated that the number of annotations collected is also of key importance for studying disagreement. For instance, Poesio and Artstein [2005] showed that what they called **implicit ambiguity**–the ambiguity emerging from disagreements among annotators, rather than from annotators explicitly marking items as ambiguous–only start to emerge for the task of anaphoric annotation when at least 5 annotations per item are collected. (The precise number of annotations appears to depend on the task.). Each item in the Gimpel dataset was annotated 5 times, apart from 946 items with a much greater number of annotations, most likely tutorial items [Gimpel et al., 2011]. The percentage of items annotated for each coder ranges from 2.64% to 5.29%. Given that there are 12 possible categories, the ratio number of coders / possible categories (the coder:label ratio) is 5:12 or 0.42.

Also important is the level of agreement between these annotators; the observed agreement, computed using the Fleiss multi-annotator version of the kappa statistic [Fleiss et al., 2004] is 0.725 overall. We also note the performance of the annotator with respect to the gold label, as a way to measure the degree of alignment between the experts and the annotators. This measurement is an indicator of how much the gold stands apart from the crowd. We use accuracy for this measure and in this dataset, the average accuracy per annotator in the GIMPEL-POS dataset is 67.81%, with over 38.98% of coders falling below this average. Only about 29% of annotators have a near-gold performance, achieving 75% or more accuracy with respect to gold labels. As shown by Snow et al. [2008], the quality of the annotators in relation to the expert is an important predictor of the quality of the classifier trained from the crowd-annotated data.

**Quality of aggregated labels** We also measure the accuracy of aggregated labels with respect to the gold as it indicates how much the crowd consensus aligns with the expert label. There is substantial disagreement in this dataset: 48.09% of the items received annotations assigning them to more than one category. Majority voting accuracy with the gold label is 79.69%; the [Dawid and Skene, 1979] and MACE aggregation methods [Hovy et al., 2013] discussed later produce labels that are 79.13% and 79.83% accurate with respect to the gold respectively.

### 2.2.2 (Anaphora and) Information Status Classification

Possibly the first type of disagreement systematically studied in NLP is disagreement on anaphoric annotation (coreference). Already identified by Artstein and Poesio [Poesio and Artstein, 2005, Poesio et al., 2006], further evidence has been unearthed as part of the annotation of virtually every modern corpus of anaphoric information: ANCORA [Recasens et al., 2011], ARRAU [Poesio and Artstein, 2008, Uryupina et al., 2020], ONTONOTES [Pradhan et al., 2012], The Potsdam Commentary Corpus [Krasavina and Chiarcos, 2007], the Prague Dependency Treebank [Nedoluzhko et al., 2016] and TUBA/DZ [Versley, 2008]. Anaphora is a more complex task than POS tagging, but it is still considered a basic aspect of semantic interpretation; yet, in the course of this research, it was discovered that depending on the genre and the range of anaphoric phenomena, annotators disagreed on 12% to 40% of all mentions. Besides the examples of ambiguity as to the antecedent of an anaphoric expression discussed in the Chapter 1 and Section 2.3, this research found subjects disagreeing as to whether the nominal form *it* is anaphoric or expletive (as in *when she [Alice] thought it over afterwards, it occurred to her that she ought to have wondered about this ...*);whether a nominal introduced a new entity or referred to an old one; and more complex cases of ambiguity to the antecedent, e.g., in cases of reference to 'split antecedent', plurals and discourse deixis [Recasens et al., 2011].

Because of the complexity of adapting models of learning from disagreement to full anaphora / coreference resolution, this study was constrained to the study of disagreements on a simplified form of the task, Information Status Classification (IS), which involves identifying the information status of a noun phrase: whether that noun phrase refers to a new entity or to an already introduced entity.

**Dataset** There are different annotation schemes for annotating information status [Prince, 1981, 1992, Nissim et al., 2004, Riester et al., 2010]. The dataset used in this work, named PDIS, is extracted from a binary version of the *Phrase Detectives* 2 corpus for coreference resolution,[3] in which a simplified, binary definition of the IS task was used, derived from the annotation scheme used in *Phrase Detectives*. In PDIS, only markables classified as introducing a new entity (discourse new - DN) , or as referring to a previously introduced entity (discourse old - DO) are considered. Markables classified as expletives or as predicative are not considered, and information about coreference chains is ignored.

The *Phrase Detectives* 2 corpus appears to be the largest NLP corpus coming with multiple annotations. It consists of a total of 542 documents containing 408K tokens and about 108K markables. 497 documents were used for training and development. These documents were annotated by over 1,828 annotators producing at least 8 annotations per markable. There are no expert annotations for the 497 training documents. 45 documents contain both expert and crowd annotations and these documents are designated as the test set. The train, development and test data each contain 97,040, 4753 and 5,855 markables respectively.

**Annotations and annotators** The full *Phrase Detectives* 2 corpus contains a total of 2,235,664 judgments, for an average of 20.6 annotations + validations per item. After restricting the judgments to only the binary DN/DO labels, and excluding validations[4], the average number of annotations was 11.87 per item for the PDIS binary subset (or 7.01 if only one annotation is counted for each annotator). The average observed agreement per item is 0.809. Each coder annotated 413.75 items on average, and the average coder accuracy is 78.13%. At least 71.25% of coders have an accuracy of 75% or more.

**Quality of aggregated labels** In PDIS, considering only the subset of data for which gold labels are available, the labels aggregated using Majority Voting are 89.54% accurate, whereas the labels aggregated using Dawid and Skene [1979] and MACE are 98.14 % and 97.89 % accurate respectively.

### 2.2.3 Relation Extraction and Frame Disambiguation

Another aspect of semantic interpretation for which there is extensive evidence of disagreements among annotators is relation extraction: the task of deciding, given two mentions and a segment of text (clause or sentence), whether that segment expresses one among a fixed number of relations between the entities referred to by those mentions. This was one of the two tasks studied most extensively in the CrowdTruth project [Aroyo and Welty, 2015, Dumitrache et al., 2018b, 2019]. Aroyo and Welty [2015] during examples encountered in projects for crowdsourcing medical relation extraction such as (2.1).

[3] The *Phrase Detectives* 2 corpus is freely available from the LDC and from `https://github.com/dali-ambiguity`.

[4] super-judgements aimed at validating or invalidating initial judgements [Poesio et al., 2019]

(2.1) GADOLINIUM AGENTS used for patients with severe renal failure show signs of NEPHROGENIC SYSTEMIC FIBROSIS.

Annotators asked to label the relation between underlined pairs with one of the UMLS relations systematically disagreed on whether pairs such as the one in the example were instances of the *cause* (strict sufficient causality) relation or the *side-effect* (possibility of a condition arising) relation. Again, both experts and novice annotators were unable to systematically make the distinction.

Two types of relation extraction were studied in the project: Medical Relation Extraction (MRE), the application to medical texts, and Frame Disambiguation, the version of the task in which the repertoire of relations is provided by FrameNet [Dumitrache et al., 2019]. This research focuses on MRE.

**Dataset** In this research, Dumitrache *et al.* created a dataset of 3,984 English sentences extracted from PubMed article abstracts for medical relation extraction centered on two main relations, the *cause* and *treat* relations, that have been processed with disagreement analysis to capture ambiguity. The sentences were sampled from the set collected by Wang and Fan [2014] using distant supervision [Mintz et al., 2009].

Dumitrache et al. [2018a] collected expert-annotations for a randomly sampled set of 975 sentences from the distant supervision dataset with each sentence being annotated by a single expert. The annotation task involved deciding whether or not the UMLS seed relation discovered by distant supervision was existent between two highlighted terms in a given sentence [Dumitrache et al., 2018a]. The crowdsourcing was carried out using so-called **disagreement-aware crowdsourcing** [Aroyo and Welty, 2015]. For every sentence, the crowd was asked to choose any number of relations from 14 possible relations, (including 'other' and 'none') applicable to the highlighted terms in the sentence.[5]

For comparability with their research. Only the subset of the cause relation with gold labels was studied for this research. The task was reframed as a binary classification task as done by Dumitrache et al. [2018a]. The gold label for each sentence given the highlighted terms is 1 if the expert agreed that the *cause* relation was existent and 0 otherwise. Similarly, for each annotator who annotated the sentence, the assigned label is 1 if the annotator selected the *cause* amongst his/her choices and 0 otherwise.

**Annotations and annotators** Each of the 975 sentences was annotated by at least 15 annotators (and a maximum of 30). On average, each coder annotated 5% of the items (a minimum of 0.1% and a maximum of 43.58%) and the average annotator accuracy is 76.1% (minimum of 0% and maximum of 100%). 58% of the annotators had an accuracy of 75% or more. The observed agreement per item is 0.857.

[5] The dataset by Dumitrache et al. [2018a] is available from `https://github.com/CrowdTruth/Medical-Relation-Extraction`.

**Quality of aggregated labels** Majority Voting was aggregated by counting the number of workers who selected the *cause* relation as a valid relation for the sentence. Labels aggregated using Majority Voting are 74.6% accurate with respect to the gold labels. Labels aggregated using Dawid and Skene [1979] and MACE are 76% and 76.61% respectively. Dumitrache et al. [2018a] also provide labels aggregation metrics using the CrowdTruth approach (discussed in section 3). These labels are 80.51% accurate with respect to the gold labels.

### 2.2.4 Recognizing Textual Entailment / Natural Language Inference

Another aspect of language interpretation for which there is systematic evidence of disagreement among subjects is Recognizing Textual Entailment (henceforth, RTE) [Dagan et al., 2006][6] Recognizing textual entailment / natural language inference is deciding whether the proposition conveyed by a text (the hypothesis $h$) can be inferred from another proposition (the premise $p$) [Dagan et al., 2006]. In NLP this task is typically formulated as a binary classification task, in which a pair $p/h$ is classified as True if the hypothesis can be inferred from the premise, False otherwise.

RTE attempts to model what is arguably the foundation of semantics [Cooper et al., 1996], but RTE judgments have proven hard for humans to agree on. Lalor et al. [2017] discuss examples like (2.2), in which it is not clear whether the hypothesis that the child plays / intends to play with the balloon follows from the premise that he's reaching for it and laughing.

(2.2)a. *Premise:* A young boy in a beige jacket laughs as he reaches for a teal balloon.
b. *Hypothesis:* The boy plays with the balloon.

In a recent and systematic analysis of disagreement on RTE judgments, Pavlick and Kwiatkowski [2019] found that workers disagreed on at least 20% of the $p/h$ pairs they were asked to classify, and that mixture Gaussian models generalized better to unseen examples than single-component Gaussians.

**Dataset** To study the effect of disagreements on learning RTE models, we used the classic PASCAL RTE-1 challenge dataset [Dagan et al., 2006] containing 800 text-hypothesis pairs [Dagan et al., 2006]. Crowdsourced annotations for this corpus were collected by Snow et al. [2008]; 164 annotators produced 10 annotations for each sentence-pair. Gold labels are also provided for each sentence-pair. This dataset was chosen as it's substantially larger than the datasets produced by Lalor et al. [2017] and Pavlick and Kwiatkowski [2019] and allowed us to compare our results with those of other researchers who used this dataset to test methods for learning from disagreement, in particular Jamison and Gurevych [2015][7].

[6] The term Natural Language Inference is now also used [Bowman et al., 2015, Pavlick and Kwiatkowski, 2019] but we will mainly use the term RTE given that this is the name of the dataset we used.

[7] `http://sites.google.com/site/nlpannotations`

**Annotations and annotators** In PASCAL RTE-1, each item received exactly 10 annotations from one of the 164 coders. This is a binary classification dataset, and the coder:label ratio is 10:2 (5). The average item observed agreement, computed using the Fleiss multi-annotator version of the kappa statistic [Fleiss et al., 2004], is 0.629.

Each coder annotated from 2.5% to 100%; an average of 6.09% of the items. The average accuracy per annotator is 83.70%, and only 35.37% of annotators fall below this average - 82.93% of them have near-gold performance, with an accuracy of at least 75%.

**Quality of aggregated labels** There is a substantial amount of disagreement prior to aggregation, much higher than with the POS dataset: 91.88% of the items have more than one unique interpretation. However, the alignment between aggregated silver labels and gold label is much higher than with POS. Majority voting aligns with the gold label in 90.25% of the cases, while using the Dawid and Skene [1979] and MACE aggregation methods produces labels that align with the gold labels in 92.88% and 92.63 % of the cases respectively.

### 2.2.5 Image Classification

Image classification is a very general term for the task of assigning an image to the category that best describes it among a fixed set of categories that depends on the application. Historically, it is possibly the area of AI that has given rise to the most work on methods for learning from disagreement. It provided the motivation for much work on methods for aggregating multiple expert-produced labels particularly for medical images [Dawid and Skene, 1979, Smyth et al., 1994, Whitehill et al., 2009]. More recently, researchers working on applications of this type have started to develop methods that learn classifiers directly from the labels produced by the crowd [Raykar et al., 2010, Albarqouni et al., 2016, Guan et al., 2018, Rodrigues and Pereira, 2018, Peterson et al., 2019]. We therefore considered it essential to include datasets used in this type of research for a proper assessment of methods for learning from disagreement, many of which originated from this area. Specifically, we employed two datasets, both of which extensively used in the literature.

#### LabelMe

**Dataset** LabelMe[8] [Russell et al., 2008] is a widely used, community-created image classification dataset where images are assigned to one of 8 categories: highway, inside city, tall building, street, forest, coast, mountain, and open country. Rodrigues and Pereira [2018] collected crowd labels for 10000 of the images using Amazon Mechanical Turk from 59 annotators producing at least one label for each image. In this study we used this version of LabelMe.

[8] http://labelme.csail.mit.edu/

**Annotations and annotators** Each item in the IC-LABELME dataset was annotated at least once and a maximum of 3 times, and the average number of annotations per item is 2.55. With 8 classes for this dataset, the average ratio number of coders / possible categories is 2.55:8, or 0.318. The average item observed agreement, computed using the Fleiss multi-annotator version of the kappa statistic [Fleiss et al., 2004], is 0.732.

Each coder annotated from 0.3% to 18.2%; an average of 6.09% of items. The average accuracy per annotator is 69.16%, and over 38.98% of coders fall below this average. 42.37% of the coders have an accuracy of 75% or more.

**Quality of aggregated labels** For this dataset, majority voting aggregation produces labels with 76.9% accuracy with respect to the gold labels while applying [Dawid and Skene, 1979] and MACE aggregation methods generates labels with 79.9 % accuracy and 78.3 % accuracy respectively.

### CIFAR-10

**Dataset** The CIFAR-10 dataset[9] is another image classification dataset used in current state-of-art research [Springenberg et al., 2015, Graham, 2014, Ghosh et al., 2017, Gastaldi, 2016]. The entire dataset consists of 60k 32x32 colour images in 10 categories (airplane, automobile, bird, cat, deer, dog, frog, horse, ship, and truck) with 6k images per class. There are 50k training images and 10k test images.

Recently, this dataset has also been used in research into learning from crowdsourced data. In particular, Peterson et al. [2019] collected human annotations for the test portion of the CIFAR-10 using Amazon Mechanical Turk, creating the CIFAR-10H dataset[10]. This dataset consists of 511,400 human categorization decisions over the 10k-images with an average of 50 annotations per image. This dataset was used for training training and testing using a 70:30 random split, but ensuring that the number of images per class remained balanced as in the original dataset. A subset of the CIFAR-10 training dataset (3k images) was used as the development set.

**Annotations and annotators** Each item was annotated an average of 51.1 times (a minimum of 47 and a maximum of 63). Given that there are 10 possible classes, the average coder:label ratio is 5.11. The average observed agreement per item is 0.924. Peterson et al. [2019] report that annotators with less than 75% accuracy were removed from the dataset, resulting in an average annotator accuracy of such that 100% of annotators have an accuracy of 75% or more.

**Quality of aggregated labels** Majority voting produces labels with 99.21% accuracy with respect to the gold labels while applying Dawid and Skene and MACE aggregation methods generates labels with 99.27% accuracy and 99.24% accuracy respectively.

[9] `https://www.cs.toronto.edu/~kriz/cifar.html`

[10] `https://github.com/jcpeterson/cifar-10h`

### 2.2.6 A summary of the datasets used in this study

Table 2.1 summarizes the statistics about the datasets discussed in this Section.

**Table 2.1:** Annotations and Annotators

| | POS | PDIS | MRE | RTE | IC-LABELME | IC-CIFAR10H |
|---|---|---|---|---|---|---|
| Number of items | 14,439 | 96,305 | 975 | 800 | 10,000 | 10,000 |
| Number of crowd workers | 177 | 1,741 | 304 | 164 | 59 | 2,457 |
| Number of Labels | 17 | 2 | 2 | 2 | 8 | 10 |
| Average annotations per item | 16.37 | 11.87 | 15.30 | 10.00 | 2.50 | 51.10 |
| Median annotations per item | 5 | 10 | 15 | 10 | 3 | 51 |
| Average number of items annotated per coder | 1335.48 | 381.757 | 49.08 | 48.78 | 431.69 | 200 |
| Median number of annotations per coder | 1276 | 20 | 14 | 20 | 270 | 200 |
| Average coder accuracy | 0.93 | 0.82 | 0.76 | 0.84 | 0.69 | 0.95 |
| Coder accuracy variance | 0.003 | 0.062 | 0.053 | 0.015 | 0.033 | 0.001 |
| Percentage of coders with accuracy above 0.75 | 1.00 | 0.77 | 0.58 | 0.83 | 0.42 | 1.00 |
| Average observed agreement per item | 0.73 | 0.81 | 0.86 | 0.63 | 0.73 | 0.92 |
| Average item entropy using raw distribution | 0.13 | 0.38 | 0.31 | 0.72 | 0.10 | 0.07 |
| Average item entropy (**best-performing distribution**, BDE) | 0.39 | 0.09 | 0.31 | 0.22 | 0.76 | 0.07 |

**Table 2.2:** Quality of Aggregated Labels

| | POS | PDIS | MRE | RTE | IC-LABELME | IC-CIFAR10H |
|---|---|---|---|---|---|---|
| Percentage accuracy of MV aggregated labels | 0.80 | 0.89 | 0.75 | 0.90 | 0.77 | 0.99 |
| Percentage accuracy of D&S aggregated labels | 0.79 | 0.98 | 0.76 | 0.93 | 0.80 | 0.99 |
| Percentage accuracy of MACE aggregated labels | 0.79 | 0.98 | 0.77 | 0.93 | 0.78 | 0.99 |

As the Tables show, the datasets differ under a number of dimensions, from the average number of annotations per item to the average number of annotations per coder to the accuracy of coders to the degree of confusion, measured in terms of observed agreement and of entropy. Chapters 4 and 6 experiments with how these differences are important in understanding the differences in performance between training methods on a dataset and the same training method on different datasets.

### 2.2.7 Other tasks

Although only the six tasks / datasets discussed above were studied in our experiments, judgment disagreements have been observed by virtually every major annotation project for virtually all language and vision interpretation tasks. In this Section we briefly review some of the literature on the presence of systematic disagreement in other aspects of language interpretation.

**Syntactic interpretation** Martínez Alonso et al. [2015] observed that the disagreements observed by Plank et al. [2014b] were characteristic of dependency parsing more in general, and applied the method proposed by Plank et al. [2014a], with promising results as well.

**Wordsense disambiguation and supersenses** Historically, the other early work on disagreements in NLP in addition to research on anaphoric disagreement arose from projects on wordsense annotation. Possibly the best known in this area is the seminal work by Passonneau and colleagues on wordsense disambiguation in the American National Corpus (see, e..g, [Passonneau et al., 2012, Passonneau and Carpenter, 2013]). Passonneau et al. [2012] carried out a systematic analysis of the disagreements for different types of words (nouns, verbs, and adjectives), and investigating the extent to which disagreements depended on annotator quality, instructions, and context. Further investigations of the practice of word sense annotation were carried out by Jurgens [2013]. Moreover, Martínez Alonso et al. [2016] showed that disagreement arises also in supersense tagging and they performed experiments using the method by Plank et al. [2014b] on English and Danish supersense datasets.

**Named entity resolution** The other NLP task systematically explored in the CrowdTruth project is Named Entity Resolution [Inel and Aroyo, 2017].

**Discourse structure** More disagreement is to be expected when considering tasks requiring more complex judgments, such as analyzing discourse structure. This intuition was already confirmed in early work by Stede [2008]. More recently, further evidence has been provided by work on argument structure annotation. The AURC-8 corpus [Trautmann et al., 2019] contains gold-standard annotations for argument component spans derived from crowdsourced labels. As well as disagreement over whether a span is argumentative or not, the starts and ends of argument components are often ambiguous, leading to significant disagreements between annotators. Simpson and Gurevych [2019] used a subset of the crowdsourced annotations from AURC-8, containing 8000 sentences, each with five judgements from 105 annotators.

**Sentiment analysis** Even more disagreement is to be expected with subjective tasks such as sentiment analysis. This intuition, too, is confirmed by evidence, such as the study by Kenyon-Dean et al. [2018] in support of the annotation of the McGill Twitter Sentiment Analysis corpus. Kenyon-Dean and colleagues found that over 30% of the instances in the corpus are “controversial” or “complicated” cases over which annotators disagree.

## 2.3 Sources of Disagreement - Errors, Imprecise Annotation Scheme, Ambiguity and Difficulty

In Section 2.2, we discussed the evidence of disagreement in several NLP and Computer Vision tasks. While all disagreements result in label uncertainty, some disagreement is intrinsic to the task and hence unavoidable; others are as a result of annotator or annotation interface errors and introduce noise to the data. To harness the disagreement in building machine learning models, some effort must be made to understand the nature and sources of the disagreements. Several annotation projects have defined and discussed various sources of disagreement. In this Section, we outline these sources of disagreement, noting their definitions in our context and highlighting their occurrences in the six datasets we explore in this research.

### 2.3.1 Errors

Disagreement can result from **annotator errors** or problems with the annotation interface. Several annotation projects have highlighted this source of disagreement; Nedoluzhko et al. [2016] found that 15% of annotator disagreement was as a result of annotator mistakes, Pradhan et al. [2012] attributed 25% of the disagreements in ONTONOTES to annotator error, and Plank et al. [2014b] found that while the ratio of noise:genuine ambiguity differed based on the level of confusion of label pairs, annotation errors made up about 30% of the disagreement for difficult items. Annotator error disagreement, while not informative about the task itself, provides information about the reliability of the annotator; their level of attention or their level of knowledge about the task.

Errors resulting in disagreement could also be as a result of **interface problems/limitations**. In coreference resolution annotation for example, errors in the markable extraction process (i.e. incorrectly defined span boundaries for markable noun phrases) often introduces annotation errors; annotators, unable to select the appropriate span either select a preceding antecedent in the same chain, a span which is a subset of the correct span, or annotate the markable as problematic. These differing judgements lead to unnecessary disagreements. Consider the following sentence from the *Phrase Detectives* corpus; the (automatically) extracted markable is in bold font and surrounded by square brackets:

> “Once upon a time there was a dear little girl who was loved by everyone who looked at her but [**most of all by her grandmother**]"

The only valid markable in that span is the noun phrase ‘*her grandmother*’. When presented with ‘*most of all by her grandmother*’, annotators disagree on that the most suitable - majority of the annotators marked it as a new markable, “Discourse New", while the others annotators marked it as a “Predicative". Our analysis of the corpus showed that interface limitations and problems account for a majority of the disagreement in validated *Phrase Detectives* corpus. Similarly, in the ONTONOTES, another coreference resolution corpus. interface and annotation scheme limitations account for 43% of the disagreements. As with annotator errors, disagreements resulting from interface errors are not informative about the tasks but are useful information about the annotation project.

### 2.3.2 Imprecise Annotation Scheme

**Imprecise or vague annotation schemes** also lead to annotator disagreements. Such annotation schemes may contain ill-defined classes or overlapping classes and/or may not cover all items, leaving an annotator unsure as to the best label for an item

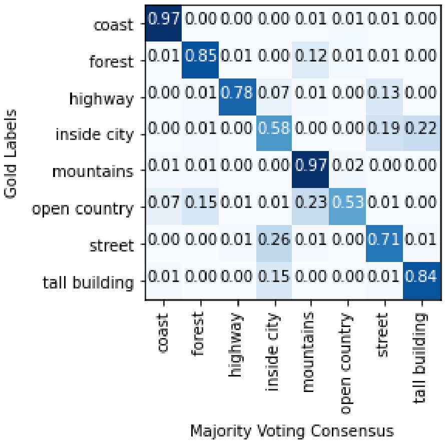
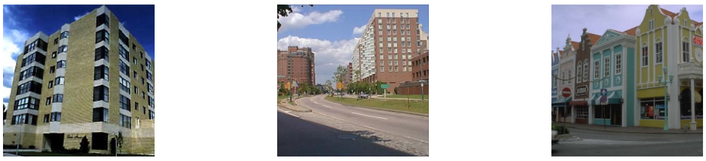


**Figure 2.1:** Confusion matrix between gold labels and majority voting consensus for LabelMe

[Nedoluzhko et al., 2016]. Russell et al.'s [2008] is a prime example of a task with an imprecise annotation scheme resulting from vague class names and descriptions. The classes *inside city*, *street* and *tall building* are not necessarily mutually exclusive so that an annotator forced to choose a category amongst them will likely be making an arbitrary choice; Figure 2.2 shows three images with buildings, each assigned a different gold label. Such examples show that even gold labels for such items are not standard but subject to the biases of the expert annotators.

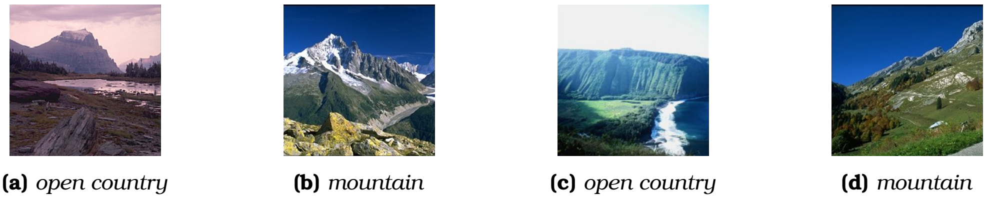


**(a)** *inside city* **(b)** *street* **(c)** *tall building*

**Figure 2.2:** Examples showing similar images from LabelMe captioned with their gold label [Russell et al., 2008, Rodrigues and Pereira, 2018]

With cases like this it is not surprising that annotators would disagree on such items. Figure 2.1 shows the confusion between majority voting consensus and gold labels for the crowdsourced LabelMe collected by Rodrigues and Pereira [2018]. The figure shows that images classified as *inside city* by the gold label are assigned to the category *tall building* by the majority 22% of the time; while images classified by gold as *street* are assigned the label *inside city* by a majority of annotators 26% of the time. This is unsurprising considering Figure 2.2. Also justifiably, images classified as *open country* by the gold are assigned the class *mountains* by the majority 23% of the time; this is justifiable as open country sometimes contain mountains. Figure 2.3 gives an illustration of this.

Similar overlap exists among other label pairs. Merging the overly fine 8 categories of LabelMe into 3 categories - (1) *coast*, (2)*inside city* + *street* + *tall building* and (3) *forest* + *mountain* + *open country* - results in majority voting aggregated labels of 95%, 18% more than the accuracy when the labels are left unmerged.



(a) *open country* (b) *mountain* (c) *open country* (d) *mountain*

**Figure 2.3:** More examples showing similar images from LabelMe, each assigned a different gold label [Russell et al., 2008, Rodrigues and Pereira, 2018]

### 2.3.3 Ambiguity

Like imprecise annotation schemes, ambiguity is a source of disagreement resulting in a multiplicity of interpretation. The difference between the two sources of disagreement lies in the fact that genuine (linguistic) ambiguity is not a consequence of a poor annotation scheme but of inherent complexity in (language) understanding and interpretation. Several studies have found **genuine ambiguity** to be a leading source of disagreement.

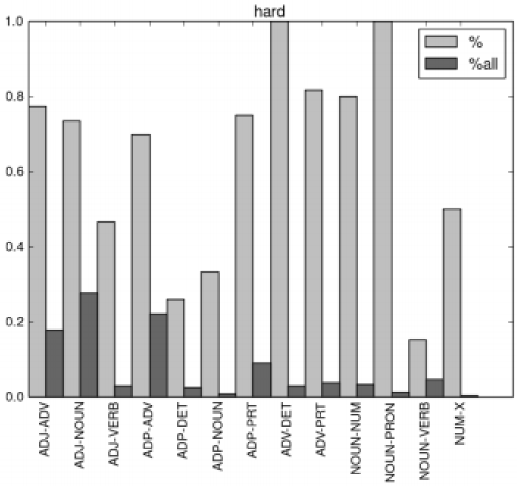


**Figure 2.4:** Figure showing the proportion of hard cases that make up 880 POS items (dark gray) and the proportion of these hard cases that are linguistically motivated (light gray) [Plank et al., 2014b]

A seminal work studying disagreement as evidence for ambiguity is the research conducted by Poesio and Artstein [2005] that studied annotation of anaphora in dialogue data. They employ 18 students to annotate the same pieces of dialogue 3.2 from the TRAINS 91 corpus by selecting the *all* valid antecedents for every markable expression the annotator perceived to be ambiguous. They found that at least 10% of the 72 markables annotated were marked *explicitly* ambiguous by at least one annotator [Poesio and Artstein, 2005]. They also found cases on *implicity ambiguity*, were markable items were not marked as ambiguous by annotators but different annotators chose different equally valid labels [Poesio and Artstein, 2005]. An analysis of some documents from the *Phrase Detectives* showed similar results. We found that while a majority of the disagreements are as a result of interface problems, 9.1% could plausibly belong to more than one coreference chain Poesio et al. [2019].

Even more evidence of disagreement as ambiguity can be found in the Part-of-Speech tagging dataset explored in this research. Plank et al. [2014b] analyse the inter-annotator disagreements and demonstrate that some disagreements are consistent across domains and languages; and certain label pairs are more confusing than others [Plank et al., 2014b]. They further employ both expert linguists to annotate 880 items from the Gimpel et al. dataset and find that a majority of the disagreement for certain label pairs stem from linguistically debatable cases [Plank et al., 2014b]. For example, Plank et al. found that all the NOUN-PRON disagreements are always linguistically debatable cases; and the same is true for 70% of the ADP-ADV disagreements. Figure 2.4 shows the the result of Plank et al.'s analysis the disagreement in involving these label pairs; the dark gray bars show the relative occurrence of a pair confusion in the dataset while light gray bars show the proportion of disagreement that is due to linguistic ambiguity. These results are further validated when 10 linguistic faculty members are asked to select the right label for 10 items in the dataset; in 8 out of 10 cases, these experts disagreed on the right tag.

It should be noted that disagreement sometimes can be **innocuous ambiguity** i.e. ambiguity that often goes undetected by a group of annotators owing to shared context; these sort of ambiguity do not impede understanding. As an illustration of the distinction between innocuous and nocuous ambiguity, consider the examples from Yang et al. [2011] in Table 2.3 (the underlined text are the possible antecedents of the pronoun in bold):

**Table 2.3:** Nocous and Innocuous examples of anaphoric ambiguity

E1: Another feature of GRASS is **its** ability to use raster, or cell, data.

E2: Table data is dumped into a delimited text file, which is sent to the remote site where **it** is loaded into the destination database.

Both examples are ambiguous as there are multiple ways of choosing the right antecedent for the pronouns. However, in the study conducted by Yang et al. [2011], 12 out of 13 readers interpreted the pronoun 'its' in example E1 as referring to 'GRASS'; whereas the readers were split almost in half regarding the antecedent selection for E2. E1 ia an example of innocuous ambiguity where despite the presence of ambiguity, a *single reading* of the sentence emerges [Yang et al., 2011] as all the readers prefer the same interpretation.

### 2.3.4 Difficulty

As seen in Section 2.3.3, in linguistic tasks like POS tagging, high disagreement might be due to genuine linguistic ambiguity, implying the validity of more than one class. As is the case with imprecise annotation schemes, ambiguity shows the gold label to be an arbitrary choice among equally valid alternatives. We make the distinction between disagreement due to genuine ambiguity and disagreement due to task difficulty based on where or not the disagreement is resolvable. While ambiguity disagreement

**Table 2.4:** 10 randomly selected polarizing items in RTE

| | Premise | Hypothesis | Gold | Observed Agreement |
|---|---|---|---|---|
| 1 | MSF was unnerved by a Taliban accusation that its members were spying for the U.S. | Taliban spies on U.S. | False | 0.44 |
| 2 | Al Thawra added, “Lahoud is well aware of that and realizes that Israeli challenges have never stopped for one moment; and that escalation will not hamper him from undertaking his national duties, relying on the support of all of Lebanon with all of its factions, as well as on the full support of Syria in order to achieve his national tasks and deliver on his commitments." | The newspaper added that regardless of the Israeli challenges, Lahoud would still be able to deliver on his duties, supported by Syria and a united Lebanon. | True | 0.44 |
| 3 | Al-Koshah’s events had surfaced after the British “Sunday Telegraph"; newspaper published last October 25 an article in which it accused Egyptian police of “;crucifying and raping Copts" | Was events Al Kusheh case emerged after the publication; newspaper Sunday Telegraph; “the British on 25 last October writes an accused in which the Egyptian police"; with steel Copts and rape of their Families | False | 0.44 |
| 4 | Seiler was reported missing March 27 and was found four days later in a marsh near her campus apartment. | Abducted Audrey Seiler found four days after missing. | True | 0.46 |
| 5 | The Bugbear virus infects computers running the Windows operating system and an unpatched version of Internet Explorer 5.5. | Virus infects thousands of computers. | False | 0.46 |
| 6 | Britney Spears is getting hitched for the second time this year - this time to a professional dancer father whose girlfriend of three years is pregnant. | Britney Spears is pregnant | False | 0.46 |
| 7 | In turn, the Editor-in-Chief of Al Jumhoria Newspaper was appointed Ambassador of Iraq to India. | Al Jumhoria is the Iraqi Ambassador to India. | False | 0.46 |
| 8 | Two Western citizens, one of whom is British, three policemen and two kidnappers were wounded in the attack that ended in the arrest of 13 kidnappers. | Wounded nationals statement one British and three police and in the attack which ended the arrest 13 | False | 0.46 |
| 9 | German Chancellor Gerhard Schroeder accused U.K. Prime Minister Tony Blair and Italian Prime Minister Silvio Berlusconi of allying with European conservative parties in a “blockade" of the German and French-backed candidate, Belgian Prime Minister Guy Verhofstadt. | Schroeder doesn’t support Vershoftstadt as a candidate. | False | 0.46 |
| 10 | Johnston is the seventh person to be killed in sectarian violence this year in Northern Ireland where the outlawed IRA is fighting to end British rule in the province. | IRA killed Johnston. | False | 0.46 |

**Table 2.5:** 10 randomly selected high agreement items in RTE

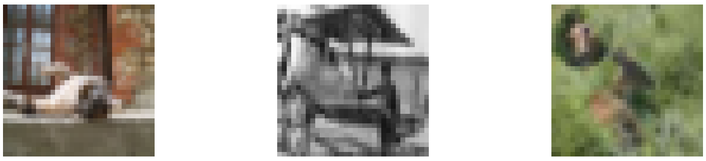
| | Premise | Hypothesis | Gold | Observed Agreement |
|---|---|---|---|---|
| 1 | Iraq has been under a stringent economic embargo since its August 1990 invasion of Kuwait and relief workers are increasingly concerned about the health of its population. | An embargo was imposed on Iraq in 1990. | True | 1.0 |
| 2 | The three-day G8 summit will take place in Scotland. | The G8 summit will last three days. | True | 1.0 |
| 3 | Kidnappings in Argentina have increased more than fivefold in the last two years, official figures show. | Argentina sees upsurge in kidnappings. | True | 1.0 |
| 4 | But even in light of this unparalleled decline, the SPD's result in the June 13 European elections is of a qualitatively different character. | European elections took place on June 13. | True | 1.0 |
| 5 | A federal jury needed just four hours to return a death sentence against Chadrick Fulks, who pleaded guilty to kidnapping and carjacking resulting in the death of an Horry County woman. | Chadrick Fulks gets the death penalty | True | 1.0 |
| 6 | The G8 summit, held June 8-10, brought together leaders of the world's major industrial democracies, including Canada, France, Germany, Italy, Japan, Russia, United Kingdom, European Union and United States. | Canada, France, Germany, Italy, Japan, Russia, United Kingdom and European Union participated in the G8 summit. | True | 1.0 |
| 7 | Crude Oil Prices Slump | Oil prices drop | True | 1.0 |
| 8 | Last July, a 12-year-old boy in Nagasaki - a city just north of Sasebo - was accused of kidnapping, molesting and killing a 4-year-old by shoving him off the roof of a car garage. | Last year a 12-year-old boy in Nagasaki was accused of murdering a four-year-old boy by pushing him off a roof. | True | 1.0 |
| 9 | Shrek 2 retained the top spot with $92.2 million over the long Memorial Day weekend, fending off the global-catastrophe tale 'The Day After Tomorrow' which debuted with $86 million, according to studio estimates Monday. | Shrek 2 retained the top spot with $92.2 million over the long Memorial Day weekend, fending off the global-catastrophe tale 'The Day After Tomorrow' which debuted with $86 million, according to studio estimates Monday. | True | 1.0 |
| 10 | Ghazi Yawar, a Sunni Muslim who lived for years in Saudi Arabia, has been picked as president of Iraq after the favored U.S. choice, Adnan Pachachi, declined to take the job. | Yawer is a Sunni Muslim. | True | 1.0 |

(and imprecise annotation scheme disagreement) result in plausible alternatives to the gold interpretation, **instance difficulty** disagreement like noise disagreement only obscure the gold label. For these difficult instances, gold labels do exist but annotator disagreements are "understandable".

While difficulty is a factor in all annotation projects, for the two of the datasets used for this research - the RTE dataset and the CIFAR-10H dataset - difficulty is a leading cause of disagreement. As an illustration of this, consider some randomly selected polarizing (high disagreement) and non-polarizing (perfect agreement) instances in RTE shown in Tables 2.4 and 2.5 respectively. We observe Table 2.4 that the examples annotators disagree on contain convoluted premises (2 and 9), convoluted hypothesis (3 and 8), or require latent information which annotators supply based on their real world biases (1,4,5,6,7,10), It is also interesting to see that the randomly selected perfect agreement instances are all $True$ according to the gold standard. Statistics show that the observed agreement for the gold $True$ class is on average higher than that of the $False$; annotators found it easier to identify entailment than to identify non-entailment. We arguably attribute these sorts of disagreement in RTE to difficulty rather than ambiguity because entailment is usually $True$ or $False$ except in cases where the hypothesis or premise is unsatisfiable. For example, instance **7** is difficult as it is unclear whether or not the newly appointed Editor-in-Chief has assumed his role; however the fact "*Al Jumhoria is the Iraqi Ambassador to India*" is either $True$ or $False$ and cannot be both.

The nature of difficulty revealed in CIFAR-10H can be observed by contrasting the examples image in Figure 2.6, for which the annotators perfectly agree, with the example images in Figure 2.5, for which observed agreement is less than 0.3. The images presented for annotation are tiny images each containing a single object among the categories under consideration[11]. In the easy cases, the objects to be identified are clearly seen; in the difficult cases the images are hard to identify. As with RTE, we largely attribute the disagreement in this task to item difficulty because even when the images are tiny or distorted, they still refer to real world objects with distinct labels.

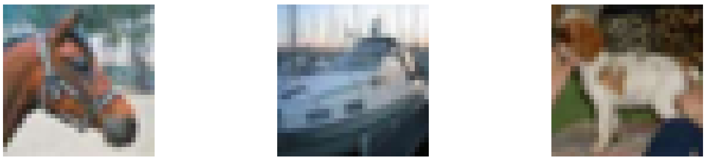
Several researchers have also shown evidence of disagreement as arising from difficulty [Beigman and Beigman Klebanov, 2009, Reidsma and Carletta, 2008, Beigman Klebanov and Beigman, 2014].



**(a)** *cat* **(b)** *airplane* **(c)** *deer*

**Figure 2.5:** Some images in CIFAR-10H with less than 0.3 observed agreement

[11] some images contain people or scenery neither of which is a category in CIFAR10



**(a)** *horse* **(b)** *ship* **(c)** *dog*

**Figure 2.6:** Some images in CIFAR-10H with perfect observed agreement

### 2.3.5 Subjectivity

Although we limit our investigations and analysis in this thesis to tasks usually assumed to be objective, this section briefly discusses disagreement arising from subjectivity. As discussed in Section 2.2.7, in tasks such as offensive language identification, annotators may disagree on whether a segment of text is offensive or not, not because of interface issues, an overlap between categories, or because they are not paying sufficient attention, nor because the items are difficult to understand, but because they have different views on whether a segment counts as offensive or not [Akhtar et al., 2019]. For instance, the Sexism dataset from Waseem [2016] consists of tweets such as (2.3) (reported by Akhtar et al. [2019]) classified by expert annotators and the crowd as either sexist or not.

(2.3) @ XXX uh... did you watch the video? one of the women talked about how it's assumed she's angry because she's latina.

Very low intercoder agreement is observed for such items, which are also flagged as being polarized by methods such as those proposed by Akhtar et al. [2019]. This is because people have different subjective views on what counts as sexist or not.

As mentioned above, this thesis focused on "objective" judgments, and thus none of the datasets studied here cover this type of disagreement; we mention it here only for completeness. But it should be clear that such cases present the most serious challenge to the gold assumption as any single label assigned to items such as (2.3) would be purely a matter of opinion.

## 2.4 Approaches to Learning from Disagreement

Current methods for learning from crowd annotations can be divided in four broad categories, summarized in Table 2.6:

1. Methods that automatically aggregate crowd annotations into (typically, one) single label for each instance. Most, although not all, of these models operate under the assumption that a single objective truth (a 'gold') exist for every instance and aim to produce the best estimate of this truth, but without requiring manual adjudication and ideally not even expert judgments. (The term **silver** truth is sometime used for these automatically aggregated labels.)

2. Methods that still assume that a gold label exists for every item, but relax the assumption that this true label is always recoverable, and use information about

disagreement to either eliminate (**filter**) items whose gold label does not appear to be recoverable due to excessive disagreement among coders (**hard** items).

3. Methods that can be used to learn a classifier directly from the crowd annotations, implicitly or explicitly assigning a score to each possible label for a given item. In this research, these approaches are collectively called ('**soft-labelling**') approaches.

4. Methods that involve training a classifier using a combination of hard labels (gold or estimated ground truth) and information from crowd annotations e.g., to weight an item by its estimated difficulty, or the ability of its annotators.

| **Category** | **Example approaches** | **Filter** | **Hard** | **Soft** |
|---|---|---|---|---|
| Aggregation of coder judgements | Dawid & Skene, CrowdTruth | | ✓ | |
| Filtering hard items | Reidsma & op den Akker | ✓ | ✓ | |
| Learning directly from crowd annotations | DLC, Soft loss, CrowdTruth | | | ✓ |
| Augmenting hard labels w/ disagreements | Plank et al., Multi-task learning | | ✓ | ✓ |

**Table 2.6:** Taxonomy of learning from disagreement. Filter: whether items are filtered out; hard labels (single ground truth); soft labels: learning from multiple annotations.

The rest of this section reviews the research that has been carried out in each of these directions. For each of these categories, a few key research projects and papers will be briefly discussed, with emphasis on the details of a method and its underlying assumption about truth. The evaluation criterion for a model will also be noted. Chapter 6 presents experiments and analysis carried out using the state-of-the-art and commonly used methods and compares these methods with the novel methods resulting from this PhD research.

### 2.4.1 Aggregating coder judgments

The simplest way to automatically aggregate a multiplicity of annotations is **Majority Voting** (MV). Using this method, the estimated label for a given item is simply the label which receives the most annotations. Majority Voting is simple to understand and implement, and can produce good results when the annotators are in agreement with each other and with the experts but it makes one key assumption that does not always apply: that all annotators are equally adept at the task. Further, Majority Voting does not take the level of difficulty of an instance into account in producing an aggregated label. These limitations are well known, and literature has been devoted to addressing them.

**Probabilistic aggregation methods** Possibly the first and definitely the most widely used method to address the limitations of Majority Voting was proposed by Dawid and Skene [1979]. Their approach (henceforth, D&S) estimates the posterior probability of a label $l_i$ for instance $i$ conditioned on the observed label $y$, the prevalence of the labels $\pi$, and the probability that an annotator assigns a particular label to an item given its actual label (this latter probability is estimated for each coder from his/her annotations):

$$p(l_i|y, \theta, \pi) \propto p(z_i|\pi)p(y|l_i, \theta)$$

Numerous other probabilistic models for estimating ground truth have been proposed after [Dawid and Skene, 1979]. Some of the most widely used include [Smyth et al., 1994, Carpenter, 2008, Whitehill et al., 2009, Hovy et al., 2013, Kamar et al., 2015, Moreno et al., 2015, Felt et al., 2015, Li et al., 2019] (see [Paun et al., 2018] for an overview and comparison of some of these models for NLP applications).[12] A model which has proven effective in many NLP applications is the simpler MACE model by Hovy et al. [2013], in which the $\theta$ parameter of D&S is replaced by a parameter $S_{ij}$ specifying the probability that coder $j$ is spamming on $i$. Some of the models, such as [Carpenter, 2008, Whitehill et al., 2009, Kamar et al., 2015], also model **item difficulty** (see below).

A non-probabilistic approach to aggregation particularly motivated by the intuition that disagreements are informative and not making the assumption that a gold truth exists was developed within the CrowdTruth project [Aroyo and Welty, 2015, Dumitrache et al., 2018c, 2019]; this approach is discussed in the next paragraph.

**The CrowdTruth approach to aggregation** The aim of the CrowdTruth project [Aroyo and Welty, 2015] was to investigate the hypothesis that 'disagreement is signal, not noise'. Research within this project led to the development of new metrics for assessing the quality of annotators, agreement on instances (a measure of item difficulty) and agreement on labels [Inel et al., 2014, Dumitrache et al., 2018c, Dumitrache, 2019] and based on these metrics, novel open and closed aggregation methods were proposed. These methods were applied to relation extraction [Dumitrache, 2019], named entity recognition [Inel et al., 2014], and a variety of other tasks with both a closed and an open number of labels [Dumitrache, 2019].

Two versions of the metrics were proposed. In both versions, the computation of the metrics is based on two basic ingredients: a **worker vector** $w_{w,i}$ recording the answers of worker $w$ on instance $i$, and a **media unit vector** $V_i$ summing up all the annotations of all the workers on instance $i$. These vectors are then used to compute quality metrics for workers, items ('media units') and classes ('annotations'):

**annotator quality:** ($WQS(i)$) - the overall agreement of one worker with the other workers;

**media unit quality:** ($UQS(u)$) - the overall worker agreement on media unit $u$; and

**'annotation quality':** ($AQS(a)$) - the overall agreement over an 'annotation', i.e., label, across all units in which it appears.

[12] A great many other surveys of aggregation methods exist. including those of Zhang et al. [2016] and Zheng et al. [2017]

In the original version of the metrics, used e.g., in Chapters 2 and 3 of Dumitrache's dissertation, these quality metrics were computed using cosine similarity to 'standards' (e.g., the 'media quality'–the extent to which an instance was a good example of a particular class–was computed by measuring the cosine similarity between that instance's media unit vector and the unit vector with a 1 for the class and 0 for all other classes). In version 2.0, all scores are mutually dependent on each other, and are therefore computed through an iterative process.

In Dumitrache's work in particular, these quality metrics were then used to assign one or more classes to an instance $i$: every label whose score for $i$ is higher than a certain (empirically established) threshold is considered a label for that instance. These metrics were shown to work better than MV in Dumitrache's work, but were not compared against other aggregation methods or other methods for using crowd annotations; in Chapter 6, this comparison is made.

The CrowdTruth project also resulted in revised versions of the standard precision / recall / F evaluation metrics Dumitrache et al. [2018c]. This metric still relies on a 'hard' label, but doesn't given the same weight to all items. Chapter 3 contains details of this metric.

**Heuristic and metric-based aggregation methods** Many heuristic-based aggregation methods were proposed (for a review see, e.g., for example Quoc Viet Hung et al. [2013], Sheshadri and Lease [2013], Daniel et al. [2018]), but none of these have been shown to outperform D&S when the estimated ground truth is compared to gold labels.

### 2.4.2 Filtering hard items

Automatic aggregation results in a single gold or estimated (silver) label for all instances in a dataset which can then be fed to a supervised classifier. Models trained using such data are usually also evaluated assuming that a single label exists for each instance in the data. In this traditional approach, even substantial disagreement on a training / testing instance does not result the removal of that instance. Several researchers have argued that information about disagreement should be used to filter the dataset: items on which there is substantial disagreement should not be used to train or evaluate models.

Reidsma and op den Akker [2008] consider inter-annotator disagreement to be an indicator of how easy or difficult an item is. They consider two ways of using disagreement to improve the performance of a classifier. The first method is to filter the data by training on the high agreement subset of the data only, i.e., treating the other items as noise. The second, softer approach is to train several classifiers, one for each annotator, and to build a 'Voting classifier' that makes a prediction when all the classifiers agree on the class label. Both methods were shown to have a high precision but low recall when evaluated on test data containing instances with varying levels of agreement.

A more systematic analysis of the effect of noisy items was carried out by Beigman-Klebanov and colleagues (see, e.g., [Beigman Klebanov et al., 2008, Beigman-Klebanov and Beigman, 2009, Beigman and Beigman Klebanov, 2009, Beigman Klebanov and Beigman, 2014]). Beigman-Klebanov *et al.* argue that low agreement on an item suggests this item is not a good example for the phenomenon at hand, as it introduces noise in a model at training time and does not allow for a fair assessment of it at test time; and should therefore be **filtered** from the training and test data, or at the very least Beigman-Klebanov and Beigman [2009], Beigman and Beigman Klebanov [2009] recommend that the low agreement (hard) instances should be separated from the high agreement (easy) cases and trained and evaluated on separately. Beigman-Klebanov and Beigman [2009] propose a model of 'hardness' that can be used to carry out this filtering or separation, but did not test this model. Beigman Klebanov and Beigman [2014] propose a simpler model based on a categorization of items ranging from 'very easy' to 'very hard' depending on the extent of disagreement, and compare the effect of selecting subsets of items at training and test time.

One noteworthy issue concerning 'hardness' is the observation by Reidsma and Carletta [2008] that not all disagreements are equally problematic for a machine learning algorithm. Disagreements are unproblematic as long as they can be viewed as random noise; they become an issue when they reveal the existence of different annotator biases, which, according to Reidsma and Carletta, is revealed by the appearance of patterns of disagreement. A proper model of 'hardness' ought to capture this finding.

Several models of 'hardness' have been explored in the literature in addition to Klebanov *et al.*'s. Arguably, the theoretically most developed approaches are the models of item difficulty incorporated in popular probabilistic aggregation models such as Carpenter's [Carpenter, 2008] and specially Whitehill's GLAD model [Whitehill et al., 2009]. Chapter 6 presenting filtering experiments using two models of item difficulty/hardness: high inter-annotator disagreement as in Reidsma and op den Akker [2008] and item difficulty computed using Whitehill's GLAD model.

### 2.4.3 Learning a classifier directly from the crowd annotations

The third category of approaches one can find in the literature includes methods that do not make the assumption that a single gold label exists or is recoverable (thus do not aim to identify a silver label, although they may weight labels according to various factors), and/or aim to capture the intuition that the distribution of labels produced by the crowd provides useful information (thus, do not attempt to filter items on which substantial disagreement was observed). Such methods, then, attempt to learn a model directly from the crowd's annotations.

Broadly speaking, one can find three varieties of this class of methods in the literature:

(1) methods that treat each annotation as a separate learning instance;

(2) methods that aggregate the annotations into a probabilistic (soft) label, then

learn directly from that distribution using a soft loss function;

(3) methods that implicitly estimate a probabilistic soft label jointly with learning a classifier.

Exemplary approaches for each of these sub-categories are discussed next.

**Multiplied examples** The first type of approach is exemplified by one of the best-known proposals in this area, by Sheng et al. [2008], who developed a **multiplied examples** approach as part of their study of the effect of repeated labelling. Sheng et al. [2008] propose that for each instance $x$, replicas of $x$ are created for each unique label $j$ assigned by the crowd to $x$. A distinct replica may be created for each annotation, or a replica may be created for each label, but weighed appropriately (e.g., it can receive a weight of $\frac{1}{|x^j|}$ or a weight of $|x^j|$, where $|x^j|$ is the number of annotations of $x$ with label $j$).

**Soft loss functions** A second but equally intuitive way to train directly from the annotations is to use the probability distributions of labels for items as soft targets in a loss function that can be used with such labels, such as cross-entropy (henceforth, CE), mean square error (MSE) or KL Divergence ((KL). This research explores this approach (see Chapters 4 and 6).

**Inducing a Classifier from Crowds** Raykar et al. [2010] pionereed the **Learning from Crowds** approach of carrying out aggregation while jointly training a model. Following the work of Dawid and Skene [1979], Raykar et al. [2010] use the Expectation Maximization algorithm to jointly learn estimated gold label, annotator reliability, and a classifier to predict whether a suspicious region on a medical image from an X-ray, CT scan, or MRI is malignant or benign. For their experiments, the multiplicity of annotations was provided several experts, radiologists and exemplified using a logistic classifier, but they also argue that the model can be used for any classifier and in a multi-class setting.

Such an extension in a deep learning setting was later proposed by, Albarqouni et al. [2016], who developed a multi-scale CNN, *AggNet*, to handle data aggregation directly as part of the learning process via an additional 'crowdsourcing layer'. Albarqouni et al. [2016] also exemplified their method also using histology image in a binary classification setting. Guan et al. [2018] also propose a neural network model for learning from medical experts (in this case, learning diabetic retinopathy severity on a 5-point scale). However, their model learns from multiple annotators (also experts) by modeling them individually with a shared net that produces unique outputs for each expert, and also learns averaging weights for combining their modeled predictions [Guan et al., 2018].

Most recently, Rodrigues and Pereira [2018] propose a similar approach to Guan et al. [2018] that they called **Deep Learning from Crowds**, that not only learns to combine the votes of multiple annotators, but also captures and corrects their biases while remaining computationally less complex than Guan et al. [2018]. Deep Learning from Crowds (henceforth: DLC) was also shown to work for binary classification, multi-class classification, regression and structured prediction problems, both in computer and NLP. In their paper, Rodrigues and Pereira [2018] show that their model outperforms existing models including [Guan et al., 2018] when evaluated against gold truth (see below).

### 2.4.4 Using both hard labels and information about disagreements

Finally, a range of methods have been proposed that assume the existence of a gold or silver hard label for each item, but also use information from the crowd annotations to improve the performance of the model. Such methods can be further subdivided into:

(1) methods that use the crowd annotations to estimate the uncertainty on the label, and use this estimate to weight the loss associated to an item; and

(2) methods that jointly learn from the hard labels and the additional information (soft labels or item difficulty).

**Plank et al.** One of the best known proposals regarding learning with disagreements in NLP, the method by Plank et al. [2014a], falls under the first sub-category. Plank et al. [2014a] compute the extent of confusion on a label from inter-annotator agreement between two expert annotators, and use that overall degree of confusion between labels to weight items while learning a part-of-speech (POS) tagging model from gold labels. They tested two different ways to quantify this label uncertainty, F1-score and tag confusion probability, finding that tag confusion probability outperformed F1 score Plank et al. [2014a].

**Sharmanska et al** A number of alternative approaches also using label confusion or item difficulty to weight the hard label have been proposed. For instance, Sharmanska et al. [2016] use inter-annotator agreement to discriminate between easy and difficult examples, but like Plank et al. [2014a], they integrate this information into their classifier as a measure of confidence in the usefulness of the data instance, instead of using it to filter the instance. They do this using a model based on Gaussian Processes in which ‘annotation ambiguities’ inform the likelihood function of the classifier regarding whether the influence of a given item should be retained, reduced or ignored. Their work was not concerned with the availability or lack of ground truth; rather, they focused on instance weighting and attempted to use disagreement to inform how much importance the learner should ascribe to each instance in the data [Sharmanska et al., 2016].

**Jointly learning from gold and disagreements** Lalor et al. [2017] proposed training algorithms in which both gold labels and soft labels were used at different times–either using gold labels for one epoch and soft labels for the next, or training using gold labels and then fine-tuning using soft labels.

As part of this research, in Chapter 5, we propose two 2 novel approaches to learning jointly from gold and crowd annotations. We propose the MTLSL model, an approach that uses a Multi-Task Learning (MTL) framework to which jointly learn to classify the hard labels (that is the gold label classes) and the soft labels, that is the coders' annotations, represented as a probability distribution [Fornaciari et al., 2021]. A similar approach MTLOA jointly learns the gold label and item confusion (as measured by observed agreement). The third approach is a pretraining approach, similar to Lalor et al. [2017] - we train the model twice, first targeting the soft labels (using the best soft loss function) and then on the gold labels, thus *pre-informing the gold*.

### 2.4.5 Coming to soft labels from another direction: Learning from noisy labels and distillation

As mentioned in the Introduction, a very active line of research in AI in general and computer vision / NLP in particular is devoted to the study of methods to learn from noisy labels above and beyond the noise due to disagreements between annotators [Mnih and Hinton, 2012, Northcutt et al., 2021]. A particular relevant line of work focuses on methods that *introduce* a measure of noise in the labels in order in order to improve generalization. Among such methods, best known is perhaps **distillation**, proposed by Hinton et al. [2015]. Distillation is a technique for transferring knowledge from a more complex, 'teacher' model to a smaller, 'pupil' model. One of the key ideas is that distillation works best when the student learns from the entire probability distribution assigned by the teacher to an item, instead of a single output.

Although there is a connection between learning from soft labels containing disagreements because they originate from human judgments, and learning from (naturally or artificial) noisy labels in general (also highlighted e.g., by Peterson et al. [2019], who compare their models for learning from disagreement with models learning via distillation), this work is outside the scope of this study, which focuses on learning from naturally generated soft labels.

# Chapter 3

# Evaluation, hard and soft

*In this Chapter, hard evaluation metrics used for this research will be mentioned and soft evaluation metrics will be proposed (cosine similarity and entropy correlation have never been used for such purposes prior to this work) along with theoretical justifications for their appropriateness - in other words, this Chapter will make a case for the use of these soft evaluation metrics in standard research. Each time the metrics are used in later chapters, references will be made to this chapter, strengthening the arguments for them.*

## 3.1 Overview

As seen in Chapter 2, there is an extensive literature concerning how to use disagreement information in learning. Much, although not all, of this work is motivated by empirical findings such as those discussed in Sections 2.3 and 2.2 suggesting that gold labels are only an idealization, at least for cognitive tasks. Yet, much less research has been devoted to the study of how to evaluate models in such circumstances, especially when it is not known what the 'true' label is.

Two forms of evaluations have been used in the literature on learning from crowdsourced data. Hard evaluation metrics such as Accuracy or F1 are traditionally used when it is assumed that a true label exists notwithstanding the disagreement between annotators. These hard metrics are briefly discussed next. More recently, however, evidence such as that presented in Section 2.2 led researchers to question the validity of evaluating models trained with data collected without assuming a gold label against test data with gold labels (e.g., using accuracy). This research takes the view that this calls for a soft evaluation approach that takes the validity of the multiplicity of opinions into account. As there is no widely accepted soft evaluation metrics, this research proposes several soft evaluation metrics. Two of these, (CE and JSD) have already used in the literature but not yet established. The others, Entropy Similarity and Entropy Correlation are novel.

The discussion of the hard evaluation metrics and the proposal of soft evaluation metrics is laid out in three sections. Sections 3.2 and 3.3 presents the hard and soft evaluation metrics - their assumptions, what they measure, and how to compute them. Section 3.4 makes comparisons between the metrics that measure the same phenomena, providing theoretical and hypothetical discussions of their appropriateness and outlining the expected behaviour of the metrics on models trained with and without the assumption of gold labels.

## 3.2 Hard evaluation

Evaluation metrics that assume a gold label are categorized as hard evaluation metrics. Hard metrics are used to evaluate a model's predictions along two criteria:

1. ***How well the model predicts gold labels when all items are treated equally***: This is the traditional 'hard' way of measuring model performance, and although many proposals for learning from multiple judgements argue against idealizing classification tasks by assuming a gold label, these proposals are still evaluated in this way. [Sheng et al., 2008, Plank et al., 2014a, Martínez Alonso et al., 2015, Sharmanska et al., 2016, Rodrigues and Pereira, 2018]. The most frequently used hard measures include percentage **Accuracy** and and class-weighted **F1**. Both of this metrics were used in this thesis. Accuracy is used to measure the proportion of "correct predictions" a model makes — i.e. the proportion of predictions the model makes in agreement with gold labels — and can be formally expressed as:

   $$Accuracy = \frac{Number\,of\,correct\,predictions}{Total\,number\,of\,predictions} \tag{3.1}$$

   F1 is especially used for evaluating models trained on imbalanced datasets – i.e. when the number of items stated to belong to each class varies widely. F1 is computed for each class and then averaged across classes. There are different ways of summing up the F1 scores across classes depending on the desired effect of the class-imbalance. A class-weighted F1 is computed by weighting F1 score of each class by the number of items in that class, and then these scores: In doing this, each class contributes to the final score, relative to its prevalence in the dataset.

   F1 is computed as follows:

   $$\text{F1} = 2 * \frac{Precision\,*\,Recall}{Precision\,+\,Recall} \tag{3.2}$$

   where

   $$Precision, P = \frac{True\,Positives}{True\,Positives\,+\,False\,Positives} \tag{3.3}$$

   and

$$Recall, R = \frac{True\,Positives}{True\,Positives\; +\; False\,Negatives} \tag{3.4}$$

True positives (henceforth *tp*) for a given class is the proportion of items that the gold standard judged as belonging to that class that the model predicts as belonging to that class. False positives (*fp*) for a class is the proportion of items judged by the gold standard as not belonging to that class which the model predicts to belong to that class. Finally, False negatives (*fn*) ) is the proportion of items judged by the gold standard as belonging to that class which the model predicts as not belonging to that class.

2. ***How well the model captures truth when items are weighed depending on disagreement***: Dumitrache et al. [2018c] propose the Crowd Truth Weighted F-measure, an alternative hard evaluation metric that assumes a gold label but weights each items contribution to the final score by the level of disagreement experienced by the crowd. The idea behind Crowd Truth Weighted F-measure (henceforth, **CT F1**) is that disagreement is a signal about the level of difficulty of an item, and that a 'fair' assessment of a model's prediction ability comes from taking this difficulty into account. The intuition is that items on which there is a lot of disagreement ('difficult' or 'confusing' items) should be weighed less than 'easy items.

   Dumitrache et al. [2018c] quantify item confusion/difficulty using a 'sentence relation score', which we will call here the 'item relation score' (*irs*) for the sake of generality. $irs(i)$ is an inverse-confusion score defined as the cosine similarity between the item annotation vector and the unit vector for the label under consideration; a higher $irs$ implies that a majority of annotators agreed with the gold labelling. Formally, $irs(i) = cos(\boldsymbol{V}_i, \hat{r})$ where $\boldsymbol{V}_i$ is the annotation unit vector discussed in Section 2.4.1 (the item's label distribution) and $\hat{r}$ is the unit vector whose dimension is the number of labels, with 0 values for all components except for the component corresponding to relation $r$.

   The $irs(i)$ is used to weigh the standard precision and recall scores, resulting in the weighted precision, $P^{'}$, and weighted recall, $R^{'}$, defined as:

$$P^{'} = \frac{\sum_i irs(i) * tp(i)}{\sum_i irs(i) * tp(i) + (1 - irs(i)) * fp(i)} \tag{3.5}$$

$$R^{'} = \frac{\sum_i irs(i) * tp(i)}{\sum_i irs(i) * tp(i) + irs(i) * fn(i)} \tag{3.6}$$

   where *tp*, *fp* and *fn* are the true positives, false positives, and false negatives defined above. The weighted f-measure, CT F1, is then defined as usual as the harmonic mean of the weighted precision, $P^{'}$, and weighted recall, $R^{'}$ as in Equation 3.2 [Dumitrache et al., 2018c].

## 3.3 Soft Evaluation

In this research, evaluation metrics that do not assume the existence of a gold label are categorized as soft evaluation metrics. This research makes a case for evaluating models that learn from multiple interpretations based on the faithfulness of their reproduction of those interpretations. No generally accepted form of soft evaluation exists if the existence of a gold label is not assumed. Therefore, we considered a variety of approaches, measuring:

1. ***How similar the distribution of labels assigned by the model to an item is to the distribution of judgments produced by the annotators for that item***: This type of evaluation captures the ability of the model to learn the probabilities of each label relative to the others for a given instance. The underlying assumption is that the item label distribution produced by the annotators is representative of the implicit ambiguity of each item.

   Given set of inputs, $\mathbf{x} = \{x_i\}_i^m$, if we define $p_{hum}(x_i)$ to be the probability distribution of the crowd annotations over the set of labels for that item and $p_\theta(x_i)$ as the probability distribution for that item produced by a model with parameters $\theta$, we measured this similarity in two ways:

   - Peterson et al. [2019] proposed to evaluate the trained models using the **Cross Entropy** (CE) function, in order to capture how confident the model is in its top prediction compared to humans and reasonableness of of its distribution over alternative categories.

     $$\textsc{ce}(p_{hum}(\mathbf{x}), p_\theta(\mathbf{x})) = \sum_{i=1}^{m} p_{hum}(x_i)\ log\ p_\theta(x_i). \tag{3.7}$$

   - **Jensen-Shannon Divergence** (JSD) [Lin, 1991] is a standard method for measuring the similarity between two probability distributions. It is based on the Kullback-Leibler divergence [Kullback and Leibler, 1951] (KL), but is symmetric and always has a finite value (see discussion below).

     The Jensen-Shannon Divergence between two probabilities $a$ and $b$ can be expressed in terms of KL divergence as follows:

     $$JSD(p_{hum}(x_i) \parallel p_\theta(x_i)) = \frac{1}{2} D_{KL}(p_{hum}(x_i) \parallel M) + \frac{1}{2} D_{KL}(p_\theta(x_i) \parallel M) \tag{3.8}$$

     where $M = \frac{p_{hum}(x_i)+p_\theta(x_i)}{2}$.
     $D_{KL}(p_{hum}(x_i) \parallel p_\theta(x_i))$ denotes the KL divergence between the two distributions and is computed as:

     $$D_{KL}(p_{hum}(x_i) \parallel p_\theta(x_i)) = p_{hum}(x_i)\ log\ \frac{p_{hum}(x_i)}{p_\theta(x_i)} \tag{3.9}$$

Using Jensen-Shannon Divergence, the similarity can be expressed as:

$$JSD(p_{hum}(\mathrm{x}), p_\theta(\mathrm{x})) = \sum_{i=1}^{m} JSD(p_{hum}(x_i) \parallel p_\theta(x_i)) \tag{3.10}$$

2. ***How well the model captures human uncertainty in its prediction***: An alternative approach to evaluating a model's ability to reproduce human judgments is to evaluate that model's ability to capture the disagreements among annotators in annotating the item, as measured using **normalized entropy**. The assumption is that the entropy of the annotators' distribution is a good measure of how confusing the annotators find the item.

   To measure the ability of a trained model $\theta$ to capture annotators' confusion, first, we compute on an item basis the normalized entropy of the probability distribution produced by the model, $H_{norm}(p_\theta(x_i))$, and the normalized entropy of the soft labels, $H_{norm}(p_{hum}(x_i))$, for each item $i$. I then compute the vectors of the entropy values over all the items, $\boldsymbol{H}_{norm\_hum}$ and $\boldsymbol{H}_{norm\_\theta}$. Finally, the model is evaluated using:

   - the cosine similarity between the two vectors, which we call the **Entropy Similarity** metric

   $$sim(\boldsymbol{H}_{norm\_hum}, \boldsymbol{H}_{norm\_\theta}) = \frac{\boldsymbol{H}_{norm\_hum} \cdot \boldsymbol{H}_{norm\_\theta}}{||\boldsymbol{H}_{norm\_hum}||||\boldsymbol{H}_{norm\_\theta}||} \tag{3.11}$$

   - the coefficient of Pearson [1896]'s correlation between the two vectors, which I call the **Entropy Correlation** metric. It is given as:

   $$corr(\boldsymbol{H}_{norm\_hum}, \boldsymbol{H}_{norm\_\theta}) \tag{3.12}$$

## 3.4 Comparisons of the evaluation metrics

In this section, the evaluation metrics for each evaluation approach are compared, providing insight into their appropriateness and expected results when used on models trained with/without assuming a gold label.

### 3.4.1 Accuracy vs F1 vs CT F1

As discussed in Section 3.2, three 'hard evaluation' metrics can be used to capture the degree of correctness of the predictions of a model/method on a task wrt the expert provided target labels: Accuracy, F1 and CT F1. The first two metrics do not take disagreement into account; the CT F1 metric on the other end weighs each item's contribution to the overall score by how confusing the annotators find that item. Hence, we have three expectations for the metrics. Firstly, we expect the relative rankings of the models to be largely similar using both Accuracy and F1, except that because the F1 metric is class weighted, we expect that in datasets with class imbalances, the

**Table 3.1:** The predictions, F1 and CT F1 of hypothetical models on a hypothetical binary task

| Model id | Model predictions | F1 | CT F1 |
|---|---|---|---|
| $m_1$ | [1, 1, 1, 1] | 1.0 | 1.0 |
| $m_2$ | [0, 0, 1, 1] | 0.5 | 0.53 |
| $m_3$ | [1, 1, 0, 0] | 0.5 | 0.28 |

rank of the methods may be different as it will be based on their performance on the majority class.

A second and third expectations about the hard metrics were set out by Dumitrache et al. [2018c,b]. To illustrate this, we'll use a simplified example. Consider a binary task with items belonging to either category $0$ or $1$ and let's assume that for 4 items in the dataset, item relation scores ($irs$) of $[0.2, 0.2, 1, 0.8]$ respectively[1] and the gold labels are $[1, 1, 1, 1]$. Then, consider three models $m_1$, $m_2$ and $m_3$. The predictions of these models and their F1 and CT F1 for class 1 on the hypothetical data subset are shown in Table 3.1. Model 1, $m_1$ is a perfect model, retrieving all the relevant items; $m_2$ retrieves the items with high $irs$ (i.e., the items for which the annotators highly agree with the gold label) and $m_3$ retrieved the low $irs$ items. Two observations can be made from the table: (1) the margin between $m_2$ and $m_1$ is narrower for CT F1 than for F1 and (2) for $m_2$, its CT F1 score is higher than its F1 score. The CT F1 score by de-emphasizing 'hard' items allows models that perform well on 'easy' items to achieve more competitive scores. This in line with the hypothesis of Dumitrache et al. [2018b]; making the assumption that for all items, the gold label is always perfectly suited/related to the item underscores the models' performance. Dumitrache et al. [2018b] further state that low(er) F1 scores for models are caused by these 'hard' items. If their hypothesis stand, we do not expect to see models behave like $m_3$ i.e. have a negative differential between their CT F1 and F1 scores.

### 3.4.2 JSD vs Cross Entropy

As mentioned in Section 3.3, the JSD function is a standard way of measuring the difference between two distributions. In coding theory, KL divergence (also known as relative entropy) is often interpreted as the number of extra bits required to send messages using the distribution, $Q$, when the optimal distribution is $P$. In the machine learning and statistical literature, KL is often used to measure the amount of information lost when $Q$ is proposed as an approximation of $P$ ($P$ typically represents the 'true' distribution and $Q$ a model's prediction). Mathematically, the relative entropy from $Q$ to $P$ (i.e. the relative entropy of $Q$ with respective to $P$) is defined as follows:

$$D_{KL}(P \parallel Q) = -\sum_{x \in \mathcal{X}} p(x) \log q(x) + \sum_{x \in \mathcal{X}} p(x) \log p(x) \tag{3.13}$$

This implies that when $p(x) = q(x)$, $D_{KL}(P \parallel Q) = 0$.

[1] Given the definition for $irs$, the more annotators agree with the gold interpretation, the higher the $irs$ score.

Cross Entropy can also be interpreted using coding theory. While KL measures the number of *extra* bits per message, Cross Entropy is the *average* or *expected* number of bits needed to send messages using $Q$ when the optimal distribution is $P$. Mathematically, the Cross Entropy of $Q$ with respect to $P$ is given as:

$$H(P, Q) = -\sum_{x \in \mathcal{X}} p(x) \log q(x) \tag{3.14}$$

Consequently, when $p(x) = q(x), H(P, Q) = -\sum_{x \in \mathcal{X}} p(x) \log p(x)$, which is the entropy of $P$ – $H(P)$. In other words, the lower bound of $H_{KL}(P, Q)$ is not *necessarily* $0$ but the entropy of $P^2$. We can also re-formulate $D_{KL}(P \parallel Q)$ as follows:

$$H(P, Q) = D_{KL}(P \parallel Q) + H(P) \tag{3.15}$$

where $H(Q)$ denotes the entropy of $P$.

The implications of these observations for the purposes of evaluating models by comparing their predicted distributions with the distribution produced by the annotators are as follows. Firstly, when a model's predictions are perfectly identical to the human's distribution, the KL divergence (and the JSD) is always zero– i.e., both KL and JSD but not CE are lower bounded at zero. Secondly, we see from equations (3.13) and (3.14) that neither KL divergence nor CE have an upper bound. The Jensen-Shannon Divergence function, however, is upper bounded at $1 \ln 2$ for the log base $e$, or 1 if using the base 2 logarithm [Lin, 1991]. This upper and lower boundedness makes JSD desirable as a metric, as the results are more easily comparable across datasets, and perhaps, hard and soft metric score can be combined (for instance by taking the sum of half of each). This is the reason we chose JSD as one of our soft evaluation metrics.[3]

Finally, the fact that JSD scores are bounded within a small range, unlike CE and KL, also means that all the results are confined with a small range than KL divergence or CE scores. This might have implications for checking the significance between results for the same dataset across models; a narrower range means the results might seem to converge to a point, making it difficult to tell, at a glance which results significantly differ from each other. For this reason, we also keep the widely known/used howbeit not bounded CE metric as a soft evaluation metric. We do, however, expect the model rankings of both CE and JSD to be largely similar.

### 3.4.3 Entropy Similarity vs Entropy Correlation

We use (normalized) entropy to measure the degree of uncertainty in the prediction of the crowd or the model for any any given item. To compare the uncertainty of a model with respect to the uncertainty of the crowd, we use Pearson correlation [Pear-

[2] In our context, the lower bound of $H_{KL}(P, Q)$ is only $0$ when a given item belongs exclusively to a single classes

[3] There has been some discussion in the literature about normalized cross entropy [Stemmer et al., 2002, Sohn, 2016], but this is not yet as widely accepted as JSD

son, 1896] and cosine similarity (see section 3). While neither satisfies the triangle of inequality and cannot be considered metrics in the mathematical sense, they both measure important relationships. With Entropy Correlation, we can measure the linear relationship between the two vectors. In other words, we can answer the question "Is the model uncertainty high when the crowd uncertainty is high, and low when crowd uncertainty is low ?". With Entropy Similarity, bounded between [0, 1], we can get a sense of how similar the vector of model entropy is to the vector of crowd entropy.

It is worth noting that cosine similarity is correlated with correlation; the more similar vectors are, the higher their correlation. As such, we expect the models' ranking by Entropy Similarity and Entropy Correlation to be largely the same. It is also worth noting that since we use the normalized version of entropy, the results using these two "metrics"[4] are comparable across datasets.

### 3.4.4 Entropy Similarity/Entropy Correlation vs Cross Entropy/JSD

The reason for using both distribution distance/divergence (using Cross Entropy/JSD) and entropy similarity/correlation (using Entropy Similarity/Entropy Correlation), can be illustrated by the following hypothetical example.

Consider a scenario where:

- $p_{hum}(x_i) = [0.8, 0.2, 0.0, 0.0]$ for a given item $i$, and
- two models $m_1$, and $m_2$ produce a $p_{\theta_1}(x_i)$ and $p_{\theta_2}(x_i)$ of $[0.6, 0.2, 0.2, 0.0]$ and $[0.0, 0.0.0.8, 0.2]$, respectively.

We can make two observations from this example. First, model $m_1$ agrees more with the crowd on where probability mass should be assigned to item $i$; and second, model $m_2$ totally disagrees with the crowd on which classes are valid for item $i$, but has the same general level of uncertainty about its prediction as the crowd does. With Cross Entropy and JSD we can capture the first type of similarity: the Cross Entropy and JSD scores for $m_1$ and $m_2$ are $(0.73, 0.33)$ and $(27.63, 1.0)$ respectively[5]. By contrast, with Entropy Similarity and Entropy Correlation we can capture the second type of similarity: $H_{norm}(p_{hum}(x_i)) = H_{norm}(p_{\theta_1}(x_i)) = 0.36$ while $H_{norm}(p_{\theta_2}(x_i)) = 0.69$.

[4] We refer to them as metrics for the purposes of this dissertation

[5] we avoid infinite values by adding by clipping the predictions using a small epsilon, 1e-12

# Chapter 4

# Soft-loss functions

*In this chapter, practical evidence is provided that learning from crowds can outperform training from experts but only under certain conditions. Theories and methods put forth in literature are evaluated using various datasets and tasks and with different characteristics, and in a variety of evaluation contexts (also, practical arguments for soft evaluation made in Chapter 2 are buttressed here). In short, research questions 3 and 4 are answered here.*

## 4.1 Introduction

In Chapter 2, we saw evidence that training AI models directly from the distributions of judgments produced by a crowd, not only provides a better account of the empirical data in NLP [Poesio and Artstein, 2005, Recasens et al., 2011, Pradhan et al., 2012, Plank et al., 2014b, Dumitrache, 2019] and computer vision [Sharmanska et al., 2016, Rodrigues and Pereira, 2018], but can also outperform models trained using ideal (gold) labels [Peterson et al., 2019]. We also discussed several methods for training directly from annotator distributions [Sheng et al., 2008, Raykar et al., 2010, Albarqouni et al., 2016, Guan et al., 2018, Rodrigues and Pereira, 2018]. As mentioned in Section 2.4.3, one intuitive way to learn from multiple judgements is to train directly on all the judgements by targeting the probability distributions of labels for each item; the training is carried out using a probability-comparing loss function. This combination of target soft labels and a probability-comparing loss function is what we call the **soft loss function**.

Recently, Peterson et al. have provided evidence of the benefits obtained using a soft loss function, applied to training a Computer Vision model for image classification and argued that such a function using CE) is the optimal loss function when the goal is to generalize well to unseen data. They showed that training using soft loss outperforms training on the gold labels. They hypothesize that the benefits are affected by the features of the dataset, and they provided an elegant demonstration that using a traditional loss function such as cross-entropy as a 'soft loss' function is optimal when the objective is to maximize performance on unseen data. However, Peterson et al. did not evaluate this proposal for other types of assessment and for other tasks. They focused on a single image classification dataset, and only compared training from human-produced probabilistic soft labels with other techniques for probabilistic label generation such as *knowledge distillation* Hinton et al. [2015]. They did not consider other methods for learning from crowd annotations, and evaluation was restricted to hard metrics and the ability to produce a distribution with minimal cross-entropy with respect to the human distribution.

In this chapter, the research into the soft-loss proposal is extended in several directions. Firstly, we carry out a systematic testing of the hypothesis that soft loss (and by extension learning from crowds) can outperform gold training using crowdsourced datasets for several AI tasks and with different characteristics, and in a variety of evaluation contexts. Precisely, we test the hypothesis on the 3 binary classification tasks - Information Status Classification using the Phrase Detectives corpus, Recognizing Textual Entailment and Medical Relation Extraction - and 3 multi-class classification tasks - POS tagging, LabelMe image classification and CIFAR10H image classification tasks - used in this research. For evaluation, we use accuracy, cross-entropy and entropy correlation; one metric for each approach to evaluation discussed in Section 3. Secondly, we carry out an extensive investigation into the hypothesis that the method used to extract a probability distribution from the raw annotations matters, the choice depending on the characteristics and amount of annotators. Finally, we carry out an analytical comparison of the soft loss and the hard loss counterparts In carrying out these experiments, this Chapter provides answers to one research question put forth in Chapter 1 namely: **RQ3**: Can models trained using multiple annotations/interpretations, without assuming gold labels, achieve similar or better performance as methods that rely on gold labels alone?

## 4.2 Methodology

### 4.2.1 Defining Soft-loss Functions

Peterson et al. [2019] proposed to train models on crowd annotated data using 'soft labels' derived from the annotations as target distributions in a cross-entropy loss function. Given some observed data $\{x_i, y_i\}_{i=1}^{n}$ at training time we want to minimize its expected loss:

$$\sum_{i=1}^{n} \sum_{c} L(f_\theta, x_i, y_i = c) p(y_i = c|x_i). \tag{4.1}$$

The second term in (4.1) using hard labels (from some consensus (adjudicated 'gold' label or aggregated 'silver' label) only yields the optimal classifier if $p(y|x)$ is 1 for a single category and 0 for all other categories, but this has been shown to be an idealization [Poesio et al., 2019, Pradhan et al., 2012, Sharmanska et al., 2016].

A more natural label categorization that factors in the uncertainty in annotation and label categorization would be the human label distribution $p_{hum}(y|x)$. Using a negative log-likelihood, this loss reduces to the cross entropy loss function CE:

$$-\sum_{i=1}^{n}\sum_{c} p_{hum}(y_i|x_i)\ \log p_\theta(y_i = c|x_i), \tag{4.2}$$

where $p_\theta(x|y)$ is obtained by applying a probability function (softmax) over the logits produced by the classifier. This combination of probabilistic soft labels with a probability-comparing loss function is what we call the soft loss function approach. In particular, we call the function expressed in Equation 4.2 the cross entropy **soft loss function** (or simply soft loss).

We explore three methods of generating $p_{hum}(y|x)$ from the crowd annotations. The first is the **standard normalization function**, also used by Peterson et al.. Peterson et al. estimate $p_{hum}(y|x)$ by applying a standard normalization function over the crowd annotations for each item. Given C classes, let $d_i$ = $[d_i^1, d_i^2, ...d_i^C]$ be a vector where some $d_i^j$ entry stores the number of times the coders chose the j-th class for the i-th training example, using normalization,

$$p_{hum}(y_i = j|x_i) = \frac{d_i^j}{\sum_a (d_i^a)} \tag{4.3}$$

This implies that any class $j$ for which the annotators provide no annotations will have a probability of 0. For datasets with numerous annotations this is a desirable effect, but for datasets with fewer annotations where some valid classes were not selected by any annotators, we hypothesize that using a **softmax** for normalization would be more appropriate, since $\exp(d_i^j) = 1$ when $d_i^j$ = 0:

$$p_{hum}(y_i = j|x_i) = \frac{\exp(d_i^j)}{\sum_a \exp(d_i^a)} \tag{4.4}$$

We hypothesize that although this transformation might introduce some noise, it is a more representative distribution for datasets with fewer and/or lower quality annotations. Furthermore, we also hypothesize that the **posterior of a probabilistic aggregation** function (prior to the use of argmax to get a single label) is a good soft label approximate for datasets with a mixed quality of annotators annotating a varying number of items. Thus, we compared soft labels generated using the standard normalization function used by Peterson et al. with soft labels generated using the softmax function and soft labels generating from the posterior distribution of two widely used probabilistic aggregation models, MACE and D&S (see Section 4.2.2).

### 4.2.2 Other Methods Tested

We test the soft loss method against hard label training methods. Hard label training involves training by targeting labels aggregated using aggregation methods. Previously, in Section 2.4.1 of Chapter 2, we discuss the theory and assumptions of 3 aggregation methods - Majority Voting, Dawid and Skene and MACE. In this section, we outline the details of training using these hard labels.

- **Majority Voting Training**: For a given data instance, majority voting selects the class with the highest number of annotations as the hard label for the instance. Majority voting training (henceforth MV training) involves training by targeting hard labels gotten by majority voting aggregation. It should be noted that aggregating soft labels generated using softmax or standard normalization results gives the majority voting aggregate for those labels.
- **Dawid and Skene Training**: As mentioned in Section 2.4, the Dawid and Skene aggregation method estimates the posterior probability of a label for a given item as a function of the prevalence of labels and the computed reliability of the coders; the label with the highest posterior probability is chosen as hard/aggregate label. Dawid and Skene training (hence D&S training) is training by targeting hard labels aggregated using Dawid and Skene aggregation method. We used a publicly available implementation of the Dawid and Skene algorithm[1] but unlike the paper which uses random initialization, we obtain initial estimates of the ground truth using majority voting.
- **MACE Training**: MACE training is training by targeting hard labels aggregated using MACE aggregation method Hovy et al. [2013], proposed as a simpler yet effective alternative to Dawid and Skene aggregation. While Dawid and Skene aggregation which learns a per-class model of annotator reliability, MACE aggregation method, only learns whether an annotator is spamming on a given instance. We use the freely available implementation of MACE provided by the authors[2]

We also compare soft loss against **gold training**, training by targeting expert-provided gold labels without the addition of crowd information.

### 4.2.3 Datasets

As mentioned from 3.1, we experiment using 6 datasets described in Chapter 2, 3 multi-class datasets - Gimpel et al.'s POS, Rodrigues and Pereira's subset of LabelMe and Peterson et al.'s CIFAR10H - and 3 binary classification datasets - PDIS, Dagan et al.'s RTE, and Dumitrache et al.'s MRE.

The smaller datasets, RTE and MRE consist of less than 1000 examples each and were trained using 10-fold cross validation. We split the larger datasets, GIMPEL-POS, PDIS, LabelMe and CIFAR10H into train:development:test sets:

- POS - GIMPEL-POS consists of over 14,000 examples with crowdsourced labels for each token. We used 12,000 examples for training and the remaining examples for testing. We used a similar dataset released by Plank et al. as a development set - this development dataset did not contain any crowd annotations.

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

[2] https://github.com/dirkhovy/MACE

- LabelMe - For LabelMe, we randomly split the 10,000 images collected by Rodrigues and Pereira into training and test data (8,882 and 1,118 images respectively) to allow for ground truth and probabilistic evaluation. 500 images from Russell et al.'s original dataset having only gold labels were used as development set.

- CIFAR10H - For IC-CIFAR10H, we used the 10,000 image CIFAR-10H dataset for training and testing using a 70:30 random split, ensuring that the number of images per class remained balanced as in the original dataset. We used a subset of the CIFAR-10 training dataset (3,000 images with only gold labels) as our development set.

### 4.2.4 Base Models

This section briefly outlines near-state-of-the-art models used for each of these tasks. This models were implemented or adapted for training using the methods outlined in Sections 4.2.1 and 4.2.2 and are available online at `https://github.com/AlexandraUma/dali-learning-with-disagreement`. The training details are included below.

**Part-of-Speech Tagging (POS) Model** For POS tagging, we implemented our own POS tagger, inspired by Plank et al. [2016], but with an attention over two kinds of input representations, the character and the word level, with each level of representation encoded using a separate RNN architecture. On a character level, each word is encoded as a sequence of characters–using a 'sequence RNN'– and the final states for each sequence of characters are used as representations. To get word-level representations, each word is encoded by passing the word embeddings through a 'context bi-RNN'; the word embeddings are initialized from pretrained Glove embeddings [Pennington et al., 2014]. Each representation is passed through a separate attention mechanism [Yang et al., 2016]. The final representation, a concatenation of these outputs, is passed through a FFN with one ReLU hidden layer and an output layer with softmax activation so that the output of the model is the probabilities for each word belonging to each of the the 12 universal POS tags.

The model was always trained for 20 epochs using the Adam optimizer [Kingma and Ba, 2015] at a learning rate of 0.001 with the the model with best development F1 saved at each epoch. This best model was used for evaluation on the test data.

**Information Status Classification Model** The model for PDIS classification[3] was developed by comparing architectures from two models: the state-of-art coreference model and the state-of-art IS classification model. The state-of-the-art model for IS classification at the time we started these experiments [Hou, 2016], was developed for the ISNOTES corpus [Markert et al., 2012, Hou et al., 2013] and achieves a performance

[3] henceforth I use PDIS to refer to both the task and the dataset, specifying the difference where necessary

of 78.9% on that corpus. The state-of-art coreference resolution system at that time [Lee et al., 2018], included a mention representation component. I developed our model by sorting the mentions using the algorithm outlined by [Hou, 2016] and a span representation similar to Lee et al. [2018] but also including the non-syntactic features from [Hou, 2016]. The model was trained for 10 epochs with training parameters set according to [Lee et al., 2018]. For each experiment, the best model based on the development set was chosen.

**Relation Extraction** For the medical relation extraction task (henceforth MRE), I fine-tuned a BERT sentence classifier [Devlin et al., 2019]. The predicted probability for a sentence is obtained by applying a softmax function over the 2D output of the classifier. The model was trained for 4 epochs using a 10-fold cross-validation at a learning rate of 2e-5. The performance of this model is much better than that of the original model by Dumitrache et al. [2018a], which only achieved an F1 of 0.638, whereas our model achieves an F1 of 0.847.

**Recognizing Textual Entailment** The RTE system described by [Dagan et al., 2006] is no longer state-of-the-art, so a new model was developed for the task. Given the small size of the dataset, the model had to be concise, with as few parameters as possible without sacrificing performance. For each item, the hypothesis and the text were each encoded using BERT [Devlin et al., 2019] and concatenated the encoded pair. This concatenation is the sentence-pair representation and is passed through a feed-forward neural network with 3 ReLU activated hidden layers and an output layer. The predicted probability for each example pair is obtained by applying a softmax function over the outputs.

The model was trained for 20 epochs using 10-fold cross-validation using Adam optimizer [Kingma and Ba, 2015] at a learning rate of 0.0001. This model trained on gold labels outperforms the model in [Jamison and Gurevych, 2015]. While the [Jamison and Gurevych, 2015] RTE achieves 51.3 micro F1, our model achieves 61.31 micro F1.

**LabelMe Image Classification Model** The model from [Rodrigues and Pereira, 2018] was replicated for learning to classify images in LabelMe (henceforth IC-CIFAR10H). Rodrigues and Pereira [2018] encoded the images using pretrained CNN layers of the VGG-16 deep neural network [Simonyan et al., 2013]. This encoding is passed into a feed-forward neural network layer with a ReLU activated hidden layer and a single output layer. Output probabilities are obtained by applying a softmax function to these outputs. Training was carried out for 50 epochs using the Adam optimizer Kingma and Ba [2015] at a learning rate of 0.001. The model with the best development result was saved and used for testing.

**CIFAR-10H Image Classification Model** The model trained for CIFAR10 image classification (henceforth IC-CIFAR10H), is the publicly available[4] Pytorch implementation of the ResNet-34A model He et al. [2016], a deep residual framework which is one of the best performing systems for the CIFAR-10 image classification. The model was trained for a total of 65 epochs divided into segments of 50, 5 and 10, using a learning rate of 0.1 and decaying the learning rate by 1e-4 at the end of every segment. The model used for the evaluation phase was the model with the best development performance.

## 4.3 Experiment Design

To achieve the goals of this chapter, the experiments are conducted in two stages. First, soft-loss models (base models discussed in Section 4.2.4 targeting soft labels) are trained for each task using probabilistic soft labels generated using (1) standard normalization and (2) softmax and evaluated with the assumption of a gold label (using accuracy). The best performing probabilistic soft label is designated as the '**true soft label**'. The second stage involves training using the other training approaches outline in Section 4.2.2 and evaluating all the models using Accuracy, Cross Entropy and Entropy Correlation metrics. For soft evaluation, the models' outputs are compared to the true soft label.

## 4.4 Results

Table 4.1 compares the two methods of generating soft labels from the crowd annotations: softmax and standard normalization. To account for non-deterministic model training effects, we average over 30 runs, except for IC-CIFAR10H and MRE which were run 10 times each due to model complexity. The mean results were reported as well as the standard deviation from the mean to show the stability of the results. The soft label generation approach that results in the best soft loss model for each task is highlighted in bold. This '**best soft loss**' is what we compare with hard loss in 4.2 and 4.3.

Tables 4.2 and 4.3 show the results of training hard and soft loss models using the various approaches for the multi-class classification and binary classification tasks respectively. Again, the models are run several times to account for non-deterministic model training effects and report the mean results were reported. We also carry significance via bootstrap sampling, following Berg-Kirkpatrick et al. [2012], Søgaard et al. [2014] to allow for a precise comparison of the methods. The in superscript rank the models in increasing order from best to worst based on significance. Models without significance differences in performance are equally ranked.

[4] https://github.com/KellerJordan/ResNet-PyTorch-CIFAR10

**Table 4.1:** Different Methods for Generating Probabilistic Labels from Crowd Annotations and their effect on Accuracy

| | POS | IS | MRE | RTE | IC-LABELME | IC-CIFAR10H |
|---|---|---|---|---|---|---|
| Standard Norm | 78.99 ± 0.36 | 90.68 ± 0.43 | **75.79 ± 0.29** | 60.24 ± 0.99 | 83.46 ± 0.82 | **66.54 ± 0.95** |
| Softmax | **79.80 ± 0.28** | 90.50 ± 0.55 | 75.27 ± 0.18 | **60.87 ± 0.84** | **84.66 ± 0.52** | 65.50 ± 1.10 |
| D&S posterior | 77.95 ± 0.61 | 92.74 ± 0.22 | 74.78 ± 0.26 | 60.51 ± 0.86 | 83.27 ± 0.76 | 65.16 ± 1.34 |
| MACE posterior | 78.27 ± 0.94 | **92.81 ± 0.26** | 75.32 ± 0.36 | 60.53 ± 0.83 | 83.53 ± 0.56 | 65.28 ± 1.02 |

## 4.5 Discussion

This section discusses several observations from the Tables 4.1, 4.2 and 4.3. Subsection 4.5.1 discusses the effects of experimenting with various approaches to generating soft labels on the accuracy of the soft loss model. Following from this discussion, 'best soft loss model' for each dataset is the best result from this table.

Subsection 4.5.2 discussions the performance of the soft loss function on multi-class datasets while Subsection 4.5.3 discusses soft loss on binary datasets. We compare the soft loss function with hard loss training on (1) silver (aggregated) labels and (2) gold labels but reserve a nuanced discussion of the different approaches to silver hard loss training for Chapter 6.

### 4.5.1 Generating Probabilistic Labels from Crowds

The results in Table 4.1 illustrate the effect on Accuracy of these different ways of obtaining the probabilistic labels. As we can see from that Table, how the probabilistic distribution is obtained does affect the results. For the MRE and IC-CIFAR10H tasks, using standard normalization to generate the soft labels yielded the most models accurate models; the softmax function yielded the best soft loss model for POS, RTE and IC-LABELME; using the posterior of the MACE aggregation model as a soft label yielded the best results for PDIS. We hypothesize that these differences can be attributed to the fact that the standard normalization function does not change the class proportions (as the softmax function does) or under-count disagreement (as the MACE and D&S posteriors do) but retains the richness of the original representation. The differences between the datasets explains why these properties of these functions matter.

We hypothesize that the standard normalization function is a better choice for high agreement datasets also having a large distribution of good quality annotations. This hypothesis is supported by the result on the tasks trained using datasets that meet this criteria: IC-CIFAR10H and MRE. For these datasets, which are characterized by a combination of (1) relatively higher observed agreement of 0.92 and 0.86 respectively (2) a median of 50 and 15 annotators per item respectively (3) annotators with an average accuracy of 0.95 and 0.76 respectively, and (4) a majority of good quality annotators (see Table 2.1), soft-loss training targeting standard normalization probabilistic labels yield the most accurate results. In general, the trend seems to be that the higher the observed agreement, the higher the accuracy of training by targeting standard normalized soft labels over targeting softmaxed soft labels (see Figure 4.1).

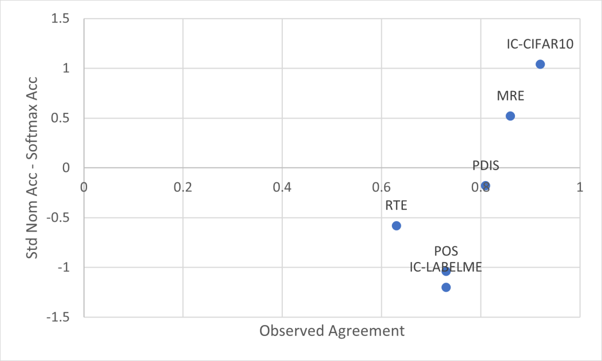


**Figure 4.1:** Graph of observed agreement against the difference in accuracy training with standard normalization (stdn) soft labels and training with softmax soft labels.

By contrast, the softmax worked best for low agreement datasets as it exacerbates disagreement and assigns a mass to to all items, even ones receiving no annotations. This affects performance with some datasets. Consider the following example from the POS dataset with the token to be tagged in bold:

> **Sentence**:"Journalists and Social Media experts alike will appreciate ***this*** spoof out of Dallas : URL"
>
> **Gold Label**: Determinant
>
> **Crowd annotations**: {Noun: 1, Pronoun:1, Adjective:1, Adposition:2}

The observed agreement for the item is 0.1, indicating that annotators found the item confusing. The standard normalization only assigns a probability to the four labels produced by annotators - {*Noun:0.2, Pronoun:0.2, Adjective:0.2, Adposition:0.4*}, while softmax assigns probabilities to these four labels {*Noun:0.11, Pronoun:0.11, Adjective:0.11, Adposition:0.30*} but also assigns a small probability of 0.04 to each of the other labels not selected by any annotators (including the $Determinant$ class). So for this low agreement item, although normalization and softmax produce distributions with the same mode (i.e., the majority vote), the softmax function unlike the standard normalization function (1) assigns a smaller mass to the modal class (which according to the gold is not the correct label for that item) and (2) assigns a small mass to the class chosen by the experts. Thus, for datasets like RTE having a relatively low observed agreement of 0.63, and datasets like POS and IC-LABELME also having relatively low agreement of 0.73 and additionally having over 11% for which the gold label did not receive any annotations, the softmax function proves to be the best option.

The PDIS dataset is a mixed bag, with an observed agreement closer to MRE and IC-CIFAR10H than to that of POS and IC-LABELME; the results reflect that. The difference in Accuracy between training with standard normalized soft labels and training with softmaxed soft labels is smallest for the PDIS dataset. However, soft-loss methods for this task benefit from using the MACE and D&S models that try to discriminate between annotators and eliminate noise, likely because of the relatively lower observed agreement (leaving room for improvement) and high number of annotations annotated per annotator (availability of ample examples to learn annotator characteristics from).

As a result of this analysis, in our experiments the ‘true soft labels’ used in the rest of this chapter is standard normalized soft labels for IC-CIFAR10H and MRE, MACE posterior for IS, and softmax soft labels for POS, RTE and IC-LABELME.

**Table 4.2:** Table showing the Accuracy (Acc), Cross Entropy (CE) and Entropy Correlation (Corr) results of soft loss and hard label training across methods for the multi-class classification tasks. Superscript indicates significance ranking of methods from best to worst - 1 being the best method for the particular task evaluated using a particular metric. Equally ranked methods are not significantly different

| | POS | | | IC-LABELME | | | IC-CIFAR10H | | |
|---|---|---|---|---|---|---|---|---|---|
| | Acc↑ | CE ↓ | Corr↑ | Acc ↑ | CE ↓ | Corr↑ | Acc↑ | CE↓ | Corr↑ |
| Gold training | **89.22**$^{1}$ | 3.34$^{3}$ | 0.41$^{3}$ | **97.21**$^{1}$ | 4.86$^{4}$ | -0.01$^{3}$ | 65.22$^{2}$ | 2.61$^{2}$ | 0.13$^{11}$ |
| MV training | 77.90$^{3}$ | 2.58$^{2}$ | 0.52$^{2}$ | 80.36$^{5}$ | 3.07$^{3}$ | 0.15$^{2}$ | 65.68$^{2}$ | 2.63$^{2}$ | 0.13$^{2}$ |
| D&S training | 77.46$^{3}$ | 2.52$^{2}$ | 0.50$^{2}$ | 83.43$^{3}$ | 2.90$^{2}$ | 0.11$^{2}$ | 65.65$^{2}$ | 2.55$^{1}$ | 0.13$^{2}$ |
| MACE training | 78.08$^{3}$ | 2.51$^{2}$ | 0.52$^{2}$ | 82.53$^{4}$ | 2.92$^{2}$ | 0.14$^{2}$ | 65.52$^{2}$ | 2.65$^{2}$ | 0.11$^{2}$ |
| Best Soft-loss | 79.80$^{2}$ | **1.35**$^{1}$ | **0.66**$^{1}$ | 84.66$^{2}$ | **1.64**$^{1}$ | **0.41**$^{1}$ | **66.64**$^{1}$ | **1.11**$^{1}$ | **0.22**$^{1}$ |

### 4.5.2 Soft Loss for Multi-class Classification

Two observations are apparent from Table 4.2: (i) soft-loss learning achieved better results at learning to predict gold labels than learning from silver labels (MV, D&S and MACE); (ii) soft-loss outperforms gold training in a single task, CIFAR-10H image classification (IC-CIFAR10H). The first observation suggests that a soft aggregation of labels from annotators that retains the uncertainty of the crowd is beneficial over a hard consensus that aims to ‘even out the noise’, irrespective of the level of expertise of the annotators or their level of disagreement. IC-CIFAR10H, annotated by highly accurate (‘expert’) annotators and POS and IC-LABELME annotated by a mixed crowd of annotators, all benefit from probabilistic soft labelling over hard labelling from crowds (see Table 2.1 for dataset characteristics).

Secondly, gold labels are usually the aggregate or adjudicated consensus of expert annotators, and as such can be very useful during learning, but as noted several times in the literature, they may present an idealization of the task which may be excessive in cases when the disagreement is real [Poesio et al., 2019, Pradhan et al., 2012, Sharmanska et al., 2016]. As seen in Figure 4.2, in a complex task like image classification disagreements in annotation can be information about the underlying difficulty of a given example. Although several annotators chose *dog* as the label for that image, *deer* and *horse* also received substantial amounts of votes, and the diverging opinions are clearly an indication of the confusing nature of the image. Probabilistic soft labels preserve label uncertainty without detracting from hard aggregated accuracy: in this case, the probabilistic soft label combines the high accuracy of majority voting with uncertainty preservation. The higher accuracy of soft-loss training compared to gold training for this task seems to suggest that particularly when the annotators are of expert quality, training using all expert annotations rather than a consensus gold label yields better results.

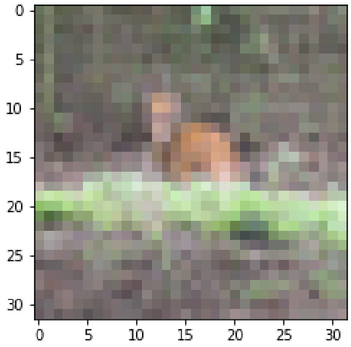


**Figure 4.2:** An example of disagreement from CIFAR10H
gold: *deer*, label with most votes (mass): *dog*, crowd counts: [*dog*:33, *deer*:13, *horse*:4];

### 4.5.3 Soft Loss for Binary Classification

One clear observation from Table 4.3 is that the soft-loss method does not significantly outperform hard loss 'silver' training for any of the datasets when evaluated using Accuracy, but rather remains on par with the best silver method. This seems to indicate that the benefits of retaining uncertainty in labelling does not apply for binary classification. This could be for one of two reasons. Firstly, it could be because there is less overlap between the labels as the classes are less fine-grained. Secondly, it could be that a different approach to soft label training is required for binary classification. Further research requiring the measurement of *expected label overlap* would be required to validate this line of reasoning.

The table also shows that although soft loss does not yield the most accurate models, soft loss models outperform all the hard loss counterparts when evaluated using soft metrics.

**Table 4.3:** Table showing the Accuracy (Acc), Cross Entropy (CE) results of soft loss and hard label training across methods for the binary classification tasks. Superscript indicates significance ranking of methods from best to worst

| | | PDIS | | | MRE | | | RTE | |
|---|---|---|---|---|---|---|---|---|---|
| | Acc↑ | CE ↓ | Corr↑ | Acc ↑ | CE ↓ | Corr↑ | Acc↑ | CE↓ | Corr↑ |
| Gold training | NA $^{5}$ | NA | NA | **84.88**$^{1}$ | 0.57$^{5}$ | 0.22$^{4}$ | **61.37**$^{1}$ | 0.77$^{2}$ | 0.04$^{2}$ |
| MV training | 90.71$^{2}$ | 0.40$^{3}$ | -0.10$^{3}$ | 75.17$^{2}$ | 0.52$^{4}$ | 0.21$^{4}$ | 60.67$^{2}$ | 0.79$^{3}$ | 0.04$^{2}$ |
| D&S training | 92.80$^{2}$ | 0.30$^{2}$ | 0.03$^{2}$ | 75.20$^{2}$ | 0.35$^{2}$ | 0.38$^{2}$ | 60.37$^{3}$ | 0.77$^{2}$ | 0.04$^{2}$ |
| MACE training | 92.90$^{1}$ | 0.30$^{2}$ | 0.03$^{2}$ | 75.15$^{2}$ | 0.46$^{3}$ | 0.27$^{3}$ | 60.55$^{2}$ | 0.80$^{3}$ | 0.02$^{2}$ |
| Best Soft-loss | **92.96**$^{1}$ | **0.27**$^{1}$ | **0.05**$^{1}$ | 75.66$^{2}$ | **0.31**$^{1}$ | **0.44**$^{1}$ | 60.87$^{1}$ | **0.74**$^{1}$ | **0.05**$^{1}$ |

## 4.6 Conclusions (Answering RQ3)

In this Chapter we make the following contributions (i) test the hypothesis that soft loss is beneficial systematically in a variety of evaluation contexts, using crowdsourced datasets for several AI tasks and with different characteristics, and comparing the results with those obtained with state-of-the-art methods for learning from crowdsourced data; and (ii) we show that the method used to extract a probability distribution from the raw annotations matters, the choice depending on the characteristics and amount of annotators.

We found that training using a cross entropy soft loss function works well not only to train models that generalize well to unseen data, as demonstrated by Peterson et al., and not only on datasets with the characteristics of IC-CIFAR10H, but in general as a method for training models from soft labels and for a variety of tasks, subject to some conditions. We also found that although this type of training does not in general outperform gold with respect to hard evaluation metrics, it does so with datasets with a substantial number of annotations per item and high quality annotations, such as IC-CIFAR10H. Also, soft-loss training systematically outperforms gold training when the objective is to achieve a model whose output mimics most closely the distribution of labels produced by the annotators, either in respect to relative ranking or in terms of uncertainty. Thus, in making these contributions, we provide an answer to the research question ‘ What is the evidence that training models using crowd annotations helps building better models in comparison with learning from expert-provided gold labels only?’

# Chapter 5

# Informing Gold Labels

*In this chapter, we answer the question 'what is the evidence that adding crowd annotation information to gold label can be beneficial over using gold models alone?' (i.e. RQ4). To answer this, two new methods for augmenting gold labels with crowd information, MTLSL and MTLOA, are introduced. Both methods are based on the Multi-Task learning paradigm. We evaluate these models using Accuracy, Cross Entropy and Entropy Correlation, comparing their results with learning using gold labels alone.*

## 5.1 Motivation

In Chapter 4, we discussed the benefits of soft loss training – training by targeting a probabilistic distribution generated from annotators – over training with gold or silver labels only. We saw that although training with gold labels might yield the most *accurate* models wrt to the gold standard, soft loss models are better at predicting labels with a distribution similar to the human (crowd) distribution as measured using metrics like cross entropy and entropy correlation. In other words, training models using crowd distribution gives them access to information that training using hard labels alone does not provide.

In this Chapter, we seek to answer the question 'Can information from crowd annotations be used in conjunction with gold labels to build better models compared to learning from gold labels only?' i.e. RQ4. To do this, we propose two novel models based on the Multi-Task (MTL) paradigm [Caruana, 1997].

## 5.2 Multi-Task Learning

In Chapter 4, we saw that more often than not, especially for noisy datasets, gold training achieves the best performance when the goal is to learn to accurately predict gold labels. However, this form of training ignores the information and signals available in crowdsourced data, evidenced in disagreement among annotators (see Chapter 2). The multi-task learning framework provides a way train a model that learns the most appropriate 'gold' label as specified by an expert or group of experts, while also informing the model about the nuances and uncertainty present in the data. MTL works by jointly learning auxiliary tasks in addition to the main task so that by leveraging the information contained in the training signals of the auxiliary tasks, the main task can be better optimized thereby improving generalization [Caruana, 1997].

For our purposes we use a hard parameter sharing architecture for our multi-task learning models. Hard parameter sharing is applied by sharing the hidden layers between all tasks, while also keeping task-specific layer(s) Ruder [2017]. This architecture, shown in Figure 5.1 has been shown to greatly reduce the risk of overvitting on the main task Ruder [2017].

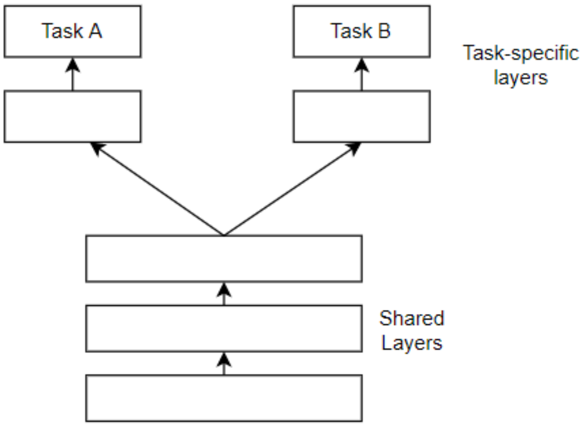


**Figure 5.1:** Hard parameter sharing MTL

For both of the MTL models proposed in this work, we use a hard parameter sharing architecture to learn the main task (gold training) and a single auxiliary task using a single task-specific layer.

The choice of an auxiliary task is rooted in the premise that the auxiliary task should be related to the main task in some way and that it should be helpful for predicting the main task. This auxiliary task is usually chosen with one or more of the following goals in mind: as an implicit form of data augmentation, to provide representation bias to the main task, to focus the attention of the model on the relevant features, as a form of eavesdropping/hinting, and/or to regularize the model by introducing an inductive bias Ruder [2017]. We propose two MTL models – MTLSL and MTLOA – that differ in the choice of auxiliary function.

### 5.2.1 Multi-Task Learning with Gold and Soft Labels (MTLSL)

The main task for this model is to learn the gold labels and the loss function for this is the negative log-likelihood loss function between the models predictions and the gold labels. The auxiliary task is to learn the probability distribution of the crowd labels i.e. probabilistic soft labels. This task was chosen to act as a regularizer to the main function; to introduce an inductive bias thereby challenges the hard assumptions of the main task. The auxiliary loss is necessarily a probability comparing loss function and for this we chose the KL-divergence as it is a natural choice to measure the difference between the prediction distribution $Q$ and the distribution of soft labels $P$. (The architecture for this model is illustrated in the Figure 5.2)

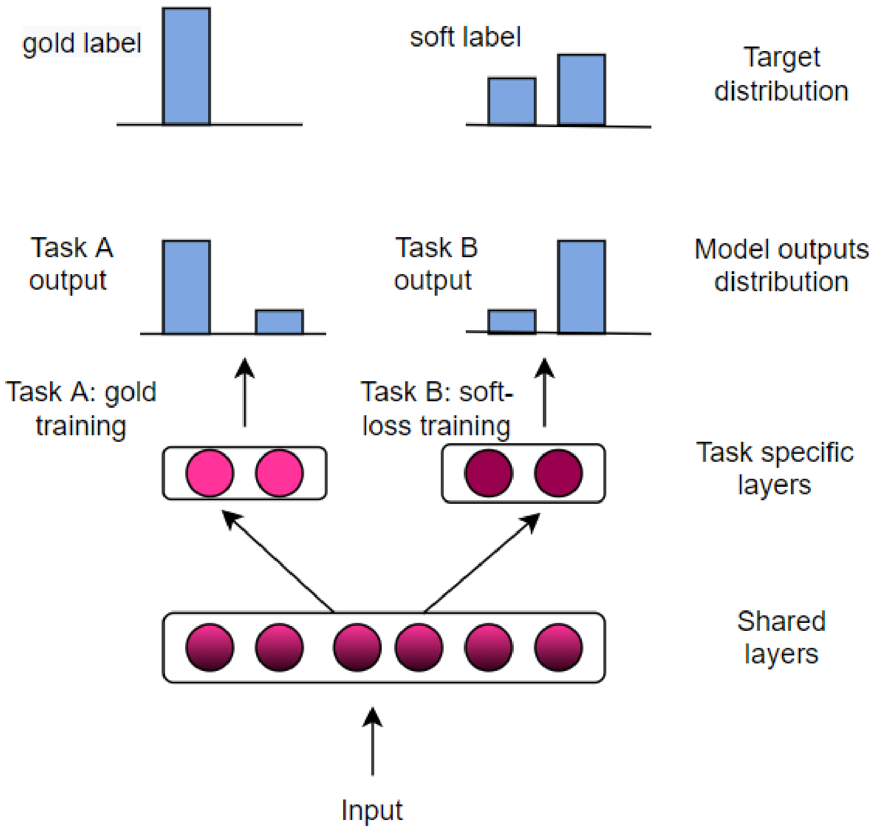


**Figure 5.2:** An illustration of the MTLSL architecture

There are two ways of using the KL-divergence function depending on the learning goal. The standard KL-divergence is:

$$D_{KL}(P||Q) = \sum_i P(i)\, log_2 \left( \frac{P(i)}{Q(i)} \right), \tag{5.1}$$

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 reverse KL-divergence is:

$$D_{KL}(Q||P) = \sum_i Q(i)\, log_2 \left( \frac{Q(i)}{P(i)} \right) \tag{5.2}$$

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.

Since the purpose of the auxiliary function is to regularize the main task, thereby reducing overfitting, we expect equation 5.2 to be more effective as it encourages the model to learn a distribution that pays attention to the classes where the annotations possibly agree. As overall loss of the main and of the auxiliary task, we backpropagate each loss in turn, the auxiliary loss warm-up loss, then the main loss.

### 5.2.2 Multi-Task Learning with Gold and Observed Agreement (MTLOA)

The auxiliary task for this MTL model is to learn the per item observed agreement. (This is illustrated in Figure 5.3). This task was chosen as an eavesdropping technique, a way to provide a hint to the model about which items to focus on during training by directly training the model to predict such items. The assumption underlying this approach is that high observed agreement is an indication of the exemplary nature of the item; in other words, high agreement items are typical of the tasks. As such, focusing on these typical examples would result in a model less attuned to the noisiness of the rare, atypical items signaled by low agreement.

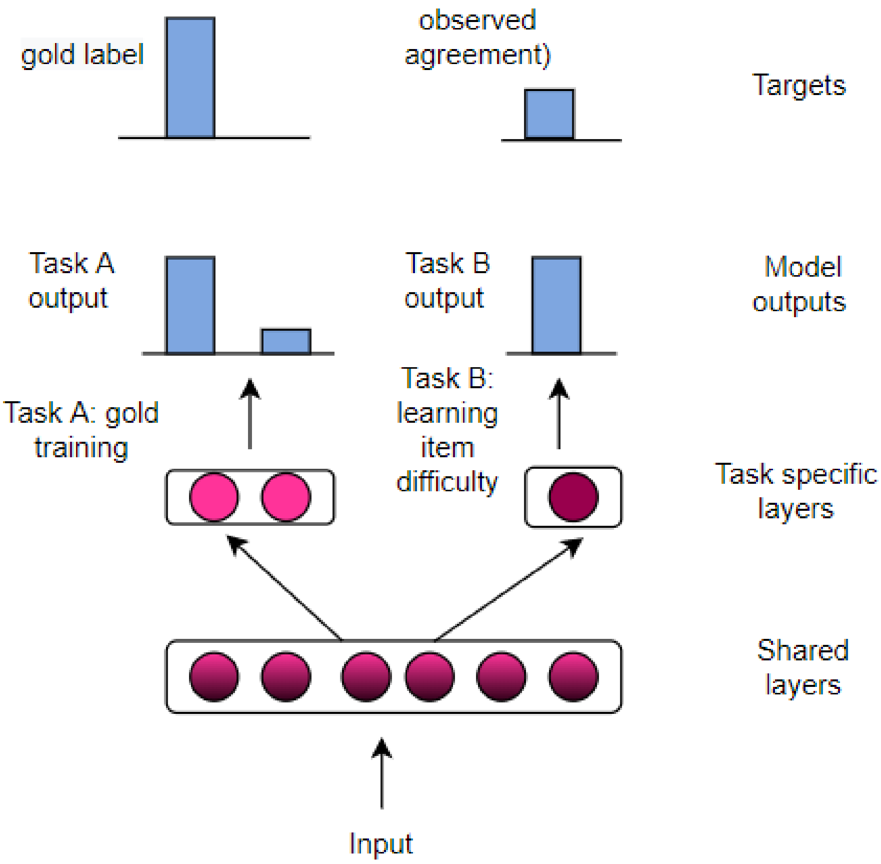


**Figure 5.3:** An illustration of the MTLOA architecture

The loss for the auxiliary function is computed by calculating the Mean-Squared Error between the predicted agreement (a Sigmoid squashed output of the task-specific layer) and the [Fleiss et al., 2004] observed agreement for all items.

### 5.2.3 Other Approaches Tested: Plank et al. [2014a]

Plank et al. propose to learn from annotator confusion by weighting the loss of each training example by the inverse of how 'confusing' the annotators find an item. They characterized the confusion using two metrics: *F1-scores* between annotators on individual POS tags, and *tag confusion probabilities* derived from confusion matrices, computed using 500 doubly-annotated tweets distinct from the dataset to be trained on [Plank et al., 2014a]. We use here the tag confusion probability which was shown by Plank et al. to have better performance than the F1-scores metrics.

To compute the tag confusion probabilities, a confusion matrix $cm$ over all the POS tags is first generated; and from this matrix, the probability of confusing two tags, $t_1$ and $t_2$ for a given item, $i$, is computed as the mean of the probability that annotator $A_1$ assigns one tag and $A_2$ another, and vice versa, i.e. $\{t_1, t_2\}$ is the mean of $cm[t_1, t_2]$ and $cm[t_2, t_1]$ [Plank et al., 2014a]. Having computed these values for every pair of tags (labels), the loss function of the classifier is augmented by multiplying the loss for each item by $1 - \{y_g, y_p\}$, where $y_g$ is the gold label for the given item, and $y_p$ is the predicted label [Plank et al., 2014a].

We adapt this idea to a multi-annotator scenario using the multiple annotations collected for each dataset. First, for each item, we compute the confusion matrix between all pairs of annotators and calculate the average confusion matrix across all the annotator, then we compute the average confusion matrix across all the items. We do this for each task independently. Using this matrix, for each we augment the loss function of each base classifier as Plank et al. do.

## 5.3 Experiment Setup

We train and evaluate the MTLSL and MTLOA models on 5 of the tasks used in this research - POS, MRE, RTE, IC-LABELME and IC-CIFAR10H - but not on PDIS as gold labels are not available for training PDIS models. We use the same base models from section 4.2.4 of Chapter 4 for these experiments and evaluate using Accuracy, Cross Entropy and Entropy Correlation as we do in Chapter 4. For MTLSL soft evaluation, we use the predictions on the auxiliary task rather than the main task. The results of these experiments are included in Section 5.4

**Table 5.1:** Accuracy results for gold and MTL methods for all tasks. Superscript indicates significance ranking of methods from best to worst - 1 being the best method for the particular task evaluated using a particular metric. Equally ranked methods are not significantly different

| | POS↑ | MRE↑ | RTE↑ | IC-LABELME↑ | IC-CIFAR10H↑ |
|---|---|---|---|---|---|
| Gold | 89.08 $^2$ | 84.88 $^1$ | 61.37 $^1$ | **97.18** $^1$ | **65.57** $^1$ |
| MTLOA | 89.26 $^2$ | 85.41 $^1$ | 61.00 $^1$ | 96.13 $^3$ | 65.23 $^1$ |
| MTLSL | **90.11** $^1$ | **85.42** $^1$ | **61.43** $^1$ | 96.82 $^2$ | 62.33 $^2$ |

## 5.4 Results

Each of Tables 5.1, 5.2 and 5.3 shows the result of mtlsl and mtloa using one evaluation metric. The results on gold training are also included in each table for comparison. As in Chapter 4, the results are the means of 10 runs for ic-cifar10h and mre and 30 runs for the other models, to account for non-deterministic model training effects. We also carry significance via bootstrap sampling, following Berg-Kirkpatrick et al. [2012], Søgaard et al. [2014] to allow for a precise comparison of the methods. The in superscript rank the models in increasing order from best to worst based on significance. Models without significance differences in performance are equally ranked.

## 5.5 Discussion

This section discusses several observations from the Tables, with Section 5.5.1 discussing the hard (accuracy) results and Section 5.5.2 and 5.5.3 discusses the soft evaluation results using cross entropy and entropy correlation.

### 5.5.1 Evaluating using Accuracy

From Table 5.1, we two key observations. Firstly, we see that the mtlsl method outperforms mtloa in 4 of the 5 tasks. This seems to indicate that a more informative auxiliary task for learning gold labels is learning the diversity of interpretations available in the crowd distribution rather than learning item difficulty as defined by observed agreement. The second key observation is that there is only one dataset for which a gold-plus method, specifically mtlsl outperforms gold training – pos. The fact that the training methods leveraging crowd information improve over gold training suggest that the crowd provides information that usefully supplements the gold labels. The pos dataset is characterized by a combination of relatively high number of judgments per item, accurate coders, relatively low observed agreement between them, and moderate 'Best Distribution Entropy'. It would seem then plausible that it is the quality, quantity *and diversity* of crowd judgments that leads to the crowd information improving over gold– which in turn suggests that the low agreement is indeed due to the fact that more than one interpretation is possible for several items in this dataset [Plank et al., 2014b].

mtlsl also yields the best accuracy results for mre and rte, though the result is not significantly better than the results on training using gold only or mtloa. The fact that we see a small but not significant improvement is, we believe, consistent with the hypothesis proposed to explain the conditions under which this happens for pos. mre has a fairly high bde, indicative of a good level of diversity, but not as high as that of pos; it has a good number of annotations per item; but the size of the dataset is likely too small to observe an effect, and coder accuracy is also fairly low. rte also has a fairly high diversity as measure by the bde.

For IC-LABELME and IC-CIFAR10H however, MTLSL results in significantly reduced hard evaluation performance with respect to gold on this dataset. We hypothesize that the reason is that the crowd annotations do not provide useful additional information for gold augmentation/regularization due to a lack of diversity. In IC-LABELME, this is observed in the low number of annotations per item (only 2.5 on average, with a over 4% of the items having only have a single annotation. For IC-CIFAR10H, however, the reason is that the crowd annotations do not provide enough diversity in comparison with the gold labels, as they appear to be drawn from the same distribution; there is little disagreement between gold labels and soft labels. This can be seen from the combination of high accuracy and high observed agreement of the crowd labels with respect to the gold, and extremely low BDE the lowest among all the datasets.

### 5.5.2 Evaluating using Cross Entropy

One thing is clear from Table 5.3; no single gold or gold-plus method achieves the best Cross Entropy results for all the tasks. For the tasks with the most diversity (as measured by BDE) – POS, MRE and IC-LABELME, the MTLSL outperforms MTLOA and gold entropy in learning the probability distribution of the crowd (as measured by Cross Entropy. This is not surprising as the auxiliary predictions of the MTLSL is to learn these crowd distributions. MTLOA always lies between gold training and MTLSL when evaluating using this metric.

**Table 5.2:** Cross entropy results for gold and MTL methods for all tasks. Superscript indicates significance ranking of methods from best to worst(Lower is better)

| | POS↓ | MRE↓ | RTE↓ | IC-LABELME↓ | IC-CIFAR10H↓ |
|---|---|---|---|---|---|
| Gold | 3.346 $^{3}$ | 0.574 $^{2}$ | **0.771** $^{1}$ | 5.159 $^{3}$ | 2.607 $^{2}$ |
| MTLOA | 3.288 $^{2}$ | 0.579 $^{2}$ | 0.796 $^{3}$ | 3.926 $^{2}$ | **2.505** $^{1}$ |
| MTLSL | **1.382** $^{1}$ | **0.569** $^{1}$ | 0.786 $^{2}$ | **1.642** $^{1}$ | 4.032 $^{3}$ |

### 5.5.3 Evaluating using Entropy Correlation

As was the case for Cross Entropy evaluation, no single method was the best across all tasks for learning to predict the perceived item difficulty of the crowd distribution as measured using Entropy Correlation. As in the other forms of evaluation, the MTLSL approach also stands out here, outperforming MTLOA and gold training in 3 of the 5 tasks examined.

**Table 5.3:** Entropy Correlation results for gold and MTL methods for all tasks. Superscript indicates significance ranking of methods from best to worst.

| | POS↑ | MRE↑ | RTE↑ | IC-LABELME↑ | IC-CIFAR10H↑ |
|---|---|---|---|---|---|
| Gold | 0.399 $^{3}$ | **0.223** $^{1}$ | 0.037 $^{1}$ | -0.016 $^{2}$ | **0.127** $^{1}$ |
| MTLOA | 0.408 $^{2}$ | 0.215 $^{1}$ | 0.035 $^{1}$ | -0.016 $^{2}$ | 0.124 $^{1}$ |
| MTLSL | **0.612** $^{1}$ | 0.120 $^{2}$ | **0.047** $^{1}$ | **0.376** $^{1}$ | 0.105 $^{2}$ |

## 5.6 Conclusions (Answering RQ4)

In response to the research question '**(a)** Can information from crowd annotations be used in conjunction with gold labels to build better models compared to learning from gold labels only? **(b)** In case the answer to (a) is positive, what is the best way of leveraging crowd information in addition to gold labels?', we introduced two novel methods, MTLSL and MTLOA, developed in the course of this research. Our results on training using these methods provides evidence that augmenting gold labels with crowd information during training *can* improve on gold performance depending on the characteristics of the dataset and the form of evaluation used. With soft evaluation using cross entropy, one of MTLSL or MTLOA significantly outperforms training on gold alone for 4 of the 5 tasks –POS, MRE and IC-LABELME – but is outperformed by gold for IC-CIFAR10H; whereas using entropy correlation, at least one gold-plus methods outperform gold training for POS and IC-LABELME but is MTLOA, is always at least on par with gold training.

With hard evaluation using accuracy, there are three datasets in which using information from the crowd ('leveraging the soft label') in conjunction with gold or hard labels helps: significantly so with POS (1 p.p. gain), marginally with PDIS and MRE. With RTE, again there is no significant difference, and which method achieves better performance depends on the metric. But with IC-LABELME and IC-CIFAR10H, using soft labels in addition to gold hurts performance. This last point is particularly surprising at the light of the fact that with IC-CIFAR10H using crowd information only achieves much better results than using gold.

We suggested that the explanation for these hard evaluation results is that soft labels help gold label training 'when the soft label provides useful information beyond the preferred label that leads to a better model'–i.e., when the soft label helps regularizing gold label training. In order for this to happen, two conditions must hold. First of all, the decision on the best label for an item must be sufficiently complex, on average. We proposed that this complexity can be measured using average Best Distribution Entropy (BDE): if the BDE is too low, leveraging soft labels in addition to hard labels doesn't help - which is why, say, MTLSL outperforms gold with POS, but not with IC-CIFAR10H, whose BDE is nearly 0, even though coder accuracy with IC-CIFAR10H in particular is very high. In other words, where the soft labels mostly reproduce the gold standard, their informative contribution lessens.

Second, there have to be enough judgments for the soft label to be sufficiently reliable. This explains why we only see marginal improvements with IC-LABELME (too few annotations per item) and RTE (too few annotations).

# Chapter 6

# A Systematic Comparison of Approaches to Learning to Classify from Crowds

*Chapter 2 reviewed the best-known evidence about disagreements on NLP and CV tasks focusing on tasks for which substantial datasets containing such information have been created and discussing the most popular approaches to training models from dataset containing multiple judgments. Chapter 3 considered how models trained on multiple judgements could be evaluated, proposing novel evaluation approaches. Chapters 4 and 5, we proposed and explored novel methods for learning from crowds and experts - soft loss, MTLOA, MTLSL and sequential fine-tuning. In this Chapter, we systematically review all the approaches discussed in Chapter 2, experimenting with the best-known and most successful method from each approach (including our proposals). We do this by training key methods under each approach for all the tasks and evaluating these methods with all the evaluation metrics discussed in Chapter 3. The goal of this line of inquiry is to answer the research questions "Among the approaches for learning from crowds, is there an absolute best method for every task?" Armed with further results, we also revisit the suitability of various evaluation metrics for evaluating the trained models.*

## 6.1 Overview

In Chapter 4, we discussed a simple yet effective method for learning from crowd annotation – the soft loss function – with proposals and experiments on how the effectiveness of the method depends on the dataset characteristics and the method used for generating soft labels from the crowd annotations. In this Chapter, we experiment with best-known and most successful method from each approach by training and evaluating models for the six datasets discussed in Section 2.2. In carrying out these experiments and analyzing the results obtained, we address **RQ5** – "Among the approaches for learning from crowds, is there an absolute best method for every task?"–

and revisit the suitability of various metrics for evaluating the disagreement-aware models.

## 6.2 Methodology

As mentioned in Section 6.1, this Chapter is a systematic comparison of the approaches to learning with disagreement across 6 datasets and tasks – POS, PDIS, RTE, MRE, IC-LABELME and IC-CIFAR10H – having strong evidence for disagreement as evidence of uncertainty (see Section 2.2). This section contains a practical discussion of these approaches to learning from disagreement. It also contains a discussion of the base models used for the various tasks.

### 6.2.1 Approaches Tested

In this section, I outline the methods for learning from disagreement we tested, providing the essential details about how they were implemented or used. Some of these methods were proposed by this research have been discussed in Chapters 4 and 5; this Chapter outlines those methods for the purposes of the systematic comparison. The methods not previously introduced in this work are discussed in full in this section. The methods are grouped according to the same categories as in Section 2.4.

#### Aggregating coder judgments

As discussed in Section 2.4, possibly the most common approach to using the labels produced by the crowd is to go through a step in which the labels used for learning are obtained either through manual adjudication or through automatic aggregation. This process is normally based on the assumption that each item belongs to a single category, but the result of this preliminary step may also be a graded ranking of the labels.

To experiment with a method in this approach, base models are trained with loss functions targeting labels aggregated using that method. The aggregation approaches evaluated in this work include:

1. **Gold Training:**
   This is training using a single gold label per instance, usually obtained through manual adjudication of annotations produced by at least two manual annotators. (All the datasets employed provided gold labels, with the exception of PDIS, which only includes gold labels for the test data.)

2. **Majority Voting:**
   This is training using for each instance the label chosen by the majority of coders.

3. **Dawid and Skene:**
   This is training using for each instance a single label produced by choosing the

label with the highest posterior probability as assigned by the Dawid and Skene [1979] algorithm which infers a per-class model of an annotator's expertise. We used a publicly available implementation of the Dawid and Skene [1979] algorithm[1] but unlike the paper which uses random initialization, we obtain initial estimates of the ground truth using majority voting.

4. **MACE:** As a probabilistic alternative to D&S, we also tested the simpler MACE item-response model [Hovy et al., 2013] which only learns whether an annotator is spamming on a given instance. This approach was shown by Hovy et al. [2013] to result in aggregated labels of higher accuracy with respect to gold labels. We used the freely available implementation of MACE provided by the authors. [2]

5. **CrowdTruth** As a final aggregation method, we tested the quasi-probabilistic approaches developed in the CrowdTruth project, which involves computing several 'quality metrics'–annotator, item, and label–to assign a label to an instance [Dumitrache et al., 2018c].

   The quasi-probabilistic approach as used in [Dumitrache et al., 2018d,b, Dumitrache, 2019] was used for Twitter Event Identification, News Event Extraction, Sound Interpretation and Medical Relation Extraction (MRE), which we experiment with in this thesis. As discussed in Section 2.2, Dumitrache et al. [2018b] collected annotations for these tasks using disagreement-aware crowdsourcing i.e. the task was multiple choice, workers were able to choose more than one relation(label) from the 14 possible relations for each item at the same time. Because the MRE dataset used here is the one used by Dumitrache et al. [2018b], and they provide the aggregated labels,[3] we use the provided labels for the CrowdTruth experiments on the MRE task. Dumitrache et al. [2018b] generate labels from MRE crowd annotations computing the following metrics:

   (a) **worker vector,** $W_{s,i}$ : For each worker, *i*, annotating a sentence $s$, the vector cell for each relation the worker selects is marked with '1,' whereas the vector cell for the relations not selected are marked with '0'.

   (b) **sentence vector,** $V_s$: The sentence vector for each sentence is computed by summing up the worker/annotation vectors for all the workers. $V_s = \sum_i W_{s,i}$

   (c) **sentence-relation score**: The sentence-relation score is computed as the cosine similarity between the sentence vector and the unit vector for that relation, $srs(s,r) = cos(V_s, \hat{r})$, where $\hat{r}$ is a one-hot vector with size the number of relations, with '0' values in all cells except for the cell corresponding to the relation being computed for. The idea is that the higher the sentence-relation score, the more clearly the relation is expressed in the sentence; hence the lower the level of ambiguity.

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

[2] https://github.com/dirkhovy/MACE

[3] https://github.com/CrowdTruth/Medical-Relation-Extraction

(d) **sentence-relation score threshold**: This is a fixed value in the [0, 1] interval used to differentiate between a negative and positive relations for a sentence. Given a sentence-relation score threshold, $t$, sentences with an $srs$ threshold less above $t$ were given a positive label, while sentences with $srs$ below $t$ received a negative label.

Given the $srs$ score for the sentences, Dumitrache et al. [2018b] produce weighted labels for the sentences by (1) separating the sentences into negative an positive sets based on the $srs$ threshold (which they chose after experimenting with several thresholds) and (2) re-scaling the labels of sentences in the negative categories in the [-1, 0] interval. They do this because the manifold model [Wang and Fan, 2014], used in the paper required either labels in the [-1, 1] interval. For our binary MRE classifier, we assign sentences in the negative set the label '0' and sentences in the positive set the label '1'. We also experimented with using corresponding weighted labels (using the $srs$ score as weights for training) but found that it led to a slight decrease in accuracy and F1 so we report the training on the unweighted binary labels.

The annotations for the other datasets experimented with in this thesis (POS, RTE, IC-LABELME, IC-CIFAR10H and PDIS) were not collected using disagreement-aware crowdsourcing; instead, for each item to be annotated, annotators could only select one of the available categories.

Extracting a single crowd ground truth using the methodology discussed above (i.e. computing the $srs$ score and a 0.5 threshold) is equivalent to majority voting, as the label with the most annotations will still be selected as the preferred label. Thus, CrowdTruth methodology to a multi-class, multi-label scenario by using a vector with as many components as the number of labels, where the components are the $srs$ scores of the corresponding labels. (A similar approach was used by Dumitrache et al. [2019] to adapt the methods to a multi-class setting.) For this reason, we consider the Crowd Truth approach for other tasks apart from MRE as a *soft label* method.

### Filtering and weighting by perceived difficulty

A second group of methods uses information about disagreements to *exclude* or at least *weigh* instances[4]. The following methods were tested:

1. **Agreement Filtering:**
   This involves training using an aggregated label but first filtering away examples with low observed agreement [Artstein and Poesio, 2008]. This was proposed by Beigman and Beigman Klebanov [2009] but there was no specific recommendation as to what the agreement cut-off ought to be. Jamison and Gurevych [2015]

[4] Instance weighing can also be categorized in the third category 'Learning Directly from the Crowd Annotations'

tested heuristically chosen two thresholds for each task: low agreement and high agreement.

In our experiments with this approach, several cut-offs were tried and the results were the same - a decline in performance for all tasks except IC-LABELME image classification. In the end, we report results obtained by filtering items with observed agreement below the average observed agreement for that dataset (which differed from task to task, as done by Jamison and Gurevych).

The formula for computing the observed agreement was computed as in [Artstein and Poesio, 2008]. Given a set of items *I* indexed by *i*, a set of categories *K* indexed by *k*, and a set of coders *C* indexed by *c*, the observed agreement for each item, $agr_i$ is given as:

$$agr_i = \frac{1}{c_i(c_i-1)} \sum_{k=1}^{K} n_{ik}(n_{ik}-1) \tag{6.1}$$

where $n_{ik}$ is the number of times item $i$ is classified as category $k$. This formula was designed under the assumption that the C is the same for each item and this does not hold true for three of the four datasets used here. To accommodate this, $c$ is adjusted to mean the number of coders annotating the given item.

$$agr_i = \frac{1}{\binom{c_i}{2}} \sum_{k=1}^{K} \binom{n_{ik}}{2} = \frac{1}{c_i(c_i-1)} \sum_{k=1}^{K} n_{ik}(n_{ik}-1) \tag{6.2}$$

2. **Weighting by Observed Agreement**

   A soft version of filtering was also tested, which involves weighting items by their degree of item difficulty instead of removing them. The idea is to weigh difficult items less, so that the model learns to pay less attention to those items and does not overfit on items for which the labels are difficult/ambiguous.

   We tried two versions of this approach. In the first version, the loss of each item is weighted by the observed agreement of that item. Learning using MV as the aggregated label, this has the effect of possibly down-weighing items on which majority voting differs from the gold interpretation. No previous references were found for this model and this work is possibly the first use of this observed agreement weighting method.

3. **Weighting by Inverse Difficulty**

   A second version of the weighing uses the inverse-difficulty predictions generated by the Whitehill et al. [2009]'s GLAD (Generative model of Labels, Abilities, and Difficulties) aggregation model. The model uses a Maximum Likelihood algorithm to simultaneously infer the annotator expertise, image difficulty and the most probable label and was exemplified for binary image classification tasks ('male' vs 'female' image categorization and *'Duchenne'* or *'Non-Duchenne'* smile image categorization).

We implemented this model and used our implementation to make item predictions for the binary classification tasks - RTE, MRE and PDIS. During training, we weigh the loss for each item by the the item's probability of correctness, an estimate that takes image difficulty and labeler quality into account Whitehill et al. [2009]'s model.

### Learning directly from the crowd annotations

The methods grouped in this category in Section 2.4 seek to train a model directly from the annotations provided by the workers, without going first through an aggregation step. The methods evaluated under this approaches are outlined below, each one a state-of-art method exemplifying a different paradigm.

1. **Repeated labelling [Sheng et al., 2008]**
   Sheng et al. [2008] proposed to train a model directly from multiple annotations by passing each annotation as input to the network as if it was a separate item. This was done as specified for 4 of the 6 tasks - POS, IC-LABELME, RTE, and MRE. Because PDIS has over 90K markables, each annotated 7 times on average, and CIFAR10H has 10K items annotated an average of 51 times and the classification model is quite complex, treating each annotation as a separate item for these tasks becomes unfeasible. Thus with these datasets the models were fed each unique label only once, but the loss for each label was weighted by the number of times that label was chosen.

2. **Soft loss functions**
   In Chapter 4 we defined the soft loss function as training by targeting the probabilistic distribution of labels obtained from the crowd annotations (aka probabilistic soft labels) as a target and show that this approach yields state-of-art hard and soft results when tested on multi-class classification tasks with varying annotator and annotation characteristics. In this Chapter, we systematically compare the best soft loss result on each dataset from Chapter 4 with the results from other approaches learning with disagreement. In addition to the cross-entropy (CE) loss function we used in Chapter 4, we also tested other loss functions that can be used to minimize the difference between probability distributions. In particular, we tested using as loss functions Mean-Squared Error (MSE) and Kullback-Leibler [Kullback and Leibler, 1951] (KL).

3. **Inducing a Classifier from Crowds** A number of methods exist to learn a model directly from the annotations by learning to weight each annotator's contributions according to their reliability Raykar et al. [2010], Albarqouni et al. [2016], Guan et al. [2018], Rodrigues and Pereira [2018]. One of the most recent such models is the *Deep Learning from Crowds* (DLC) approach, proposed by Rodrigues

and Pereira [2018]. DLC not only learns to combine the votes of multiple annotators, but also captures and corrects their biases while remaining computationally less complex than previous methods. Rodrigues and Pereira [2018] showed that their model outperforms several existing models when evaluated against gold truth. For these reasons, we select DLC as the representative method for this kind of approach. The DLC approach involves adding a bottleneck layer, called a "crowd layer", after the output layer during training, so that the model learns how much weight to assign to each label by learning the annotator matrix. Suppose the output of a neural network model is denoted by $\boldsymbol{\sigma}$, such that $\sigma_c$ corresponds to the score assigned by the model to the input instance belonging to class c, the activation of the crowd layer for each annotator, $r$ is defined as $\boldsymbol{a}^r = f_r(\boldsymbol{\sigma})$, where $f_r$ is an annotator-specific function [Rodrigues and Pereira, 2018]. This way, the output of the crowd layer is simply the softmax of the activations [Rodrigues and Pereira, 2018]. An illustration of this is shown in Figure 6.1.

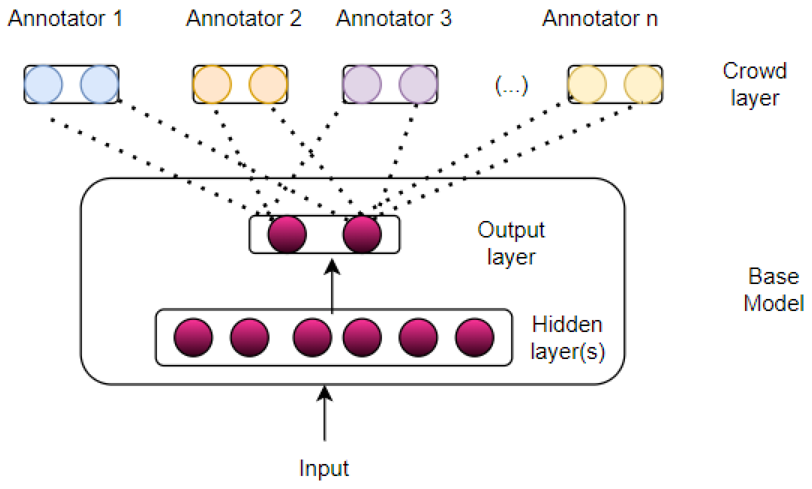


**Figure 6.1:** Label Crowd layer for the image classification task

**DLC training** involves adding a crowd layer to the base models for each task (see section 4.2.4). In particular, we used the DL-MW variant that achieved the most accurate predictions in Rodrigues and Pereira [2018]. In this variant, $f_r(\boldsymbol{\sigma})$ is defined as $\boldsymbol{W}^r(\boldsymbol{\sigma})$ where $\boldsymbol{W}^r$ is an annotator-specific matrix of the estimated sensitivities and specificities of the annotators, which was initialized to an Identity matrix that is a trainable parameter of the neural network model. As Rodrigues and Pereira [2018] do, at test time the crowd layer is removed and evaluation was carried out using the softmax output of each base model. This approach involves adding a bottleneck layer, called a "crowd layer", after the output layer during training, so that the model learns how much weight to assign to each label by learning the annotator matrix (see Section 3 of Chapter 4). As this approach was the state-of-art approach for learning from crowds at the time of these experiments, we compared the DLC method with the novel soft loss method in Chapter 4. The results of this comparison are included here for a more systematic comparison across approaches.

**Using both gold labels and information about disagreement**

This approach, introduced in Chapter 5, is comprised of methods that learn a classifier using both the gold labels and additional information extracted from disagreements. These methods were introduced in Chapter 5 but in this section, an in-depth comparison is made between them and other approaches to learning from crowds in other to provide insight into the research question 5, 'what is the best method for learning from crowds?'. The methods include:

1. **Plank Style Weighting [Plank et al., 2014a]** learning with inter-annotator agreement loss,
2. **Multi-task Learning with soft labels [Fornaciari et al., 2021]** learning from gold labels and soft labels using a multi-task learning paradigm, and
3. **Multi-task Learning with observed agreement** jointly learning ground truth as you learn a item difficulty as specified by observed agreement using the multi-task learning.

### 6.2.2 Base Models

The experiments in learning with disagreement involve extending base models for the various tasks to learn from multiple judgements, as specified by the particular approach. [5] For example, using Majority Voting approach to learning from crowd discussed in Section 2.4, the Majority Voting training approach implies targeting hard labels aggregated by majority voting using a negative likelihood loss function. In this section, I describe the base models used for the various tasks.

The base models used for the experiments in this Chapter were discussed in Chapter 4.

## 6.3 Results

As anticipated, in this Chapter, we carry out an in-depth analysis of the approaches to training from disagreement discussed in Section 2.4 by using them to train the models for the datasets discussed in Section 2.2, and evaluating the resulting models using the metrics discussed in Section 3. The results are summarized here, and analyzed in greater detail in the next two.

Tables 6.1 to 6.7 present the results for all the methods on all the tasks using a distinct evaluation metric. For comparison, we also include the results obtained by

[5] Code for this is available online at `https://github.com/AlexandraUma/dali-learning-with-disagreement`

Table 6.1: Accuracy results for all the methods on all the tasks

| | | POS | PDIS | MRE | RTE | IC-LABELME | IC-CIFAR1OH |
|---|---|---|---|---|---|---|---|
| 1 | Gold | 89.08$^{17}$ | NA | 84.88 | 61.37 | **97.18** | 65.57$^{7}$ |
| 2 | Majority Voting Silver | 78.09$^{9}$ | 90.71$^{5}$ | 75.17$^{12}$ | 60.67$^{1}$ | 80.23$^{4}$ | 65.31$^{7}$ |
| 3 | Dawid and Skene Silver | 77.67$^{4}$ | 92.80 | 75.20$^{12}$ | 60.37$^{7}$ | 83.58$^{9}$ | 65.65$^{7}$ |
| 4 | MACE Silver | 78.08$^{9}$ | 92.90 | 75.15$^{12}$ | 60.55$^{1}$ | 82.53$^{6}$ | 65.52$^{7}$ |
| 5 | Crowd Truth Silver | 79.33$^{7}$ | 91.30$^{6}$ | 75.17$^{12}$ | 60.37$^{13}$ | 84.50$^{13}$ | 64.09$^{4}$ |
| 6 | Sheng Repeated Labelling | 79.23$^{7}$ | 92.11$^{15}$ | 75.66$^{12}$ | 60.01$^{2}$ | 83.46$^{9}$ | **68.46** |
| 7 | CE loss + probabilistic labels | 79.80$^{1}$ | 92.86 | 75.55$^{12}$ | 60.87 | 84.66$^{13}$ | 66.54$^{10}$ |
| 8 | KL loss + probabilistic labels | 79.96$^{1}$ | 92.86 | 75.53$^{12}$ | 60.68$^{1}$ | 84.73$^{13}$ | 66.58$^{10}$ |
| 9 | MSE loss + probabilistic labels | 79.20$^{7}$ | 92.90 | 75.50$^{12}$ | 60.70$^{16}$ | 84.21$^{7}$ | 63.49$^{4}$ |
| 10 | DLFC | 77.87$^{4}$ | 92.82 | 74.67$^{4}$ | 59.75$^{5}$ | 83.69$^{9}$ | **68.25** |
| 11 | MV + OA Hard Filter | 72.20$^{3}$ | 68.51$^{12}$ | 74.85$^{4}$ | 54.77$^{12}$ | 86.05$^{12}$ | 63.98$^{4}$ |
| 12 | Gold + OA Hard Filter | 79.84$^{1}$ | 73.28$^{14}$ | 83.18$^{1}$ | 55.77$^{10}$ | 94.60$^{16}$ | 63.54$^{4}$ |
| 13 | MV + OA Weighting | 78.17$^{9}$ | 90.44$^{6}$ | 75.29$^{12}$ | 61.04 | 85.54$^{12}$ | 65.99$^{10}$ |
| 14 | MV + WH Weighting | NA | 90.31$^{5}$ | 75.25$^{12}$ | 58.76$^{6}$ | NA | NA |
| 15 | Gold + Plank et al weighting | 89.26$^{17}$ | 92.70 | **85.43** | 61.15 | 96.37$^{17}$ | 64.78$^{13}$ |
| 16 | MTLOA | 89.26$^{17}$ | 92.86 | 85.41 | 61.00 | 96.13$^{17}$ | 65.23$^{7}$ |
| 17 | MTLSL | **90.11** | **92.95** | 85.42 | **61.43** | 96.82$^{1}$ | 62.33$^{5}$ |

training the same architectures using gold labels (with a cross entropy loss function). In these Tables there is one row for each learning from disagreement method, and one column for each dataset used in the evaluation. Double lines are used to group closely related methods in sections, one for each of the class of methods in Section 6.2.1 (e.g., aggregation methods).

The best result for each dataset is highlighted in bold, whereas the best result among the training methods not using gold information is underlined. In each cell (i.e. for the result of a given training method on a particular dataset), we also include in superscript the row number of the method with minimum significant improvement over the method in the cell, if any. To account for non-deterministic model training effects, each model was re-run 30 times, except for (i) IS, which was only run 10 times owing to the size of the dataset and model complexity, and (ii) IC-CIFAR1OH and MRE, also run 10 times due to model complexity.

### 6.3.1 Evaluation against gold or hard labels

Tables 6.1, 6.2 and 6.3 show the results of evaluation against gold labels, using Accuracy, F1 and the weighted version of F1 developed in the Crowd Truth project, CT F1. Figures 6.2 summarizes these results by displaying for each category of methods the results obtained by the best performing approach in that category for each dataset.

The first broad conclusion we can reach from these Tables and from Figure 6.2 is that the answer to **RQ3** is in most cases negative if we use 'hard' evaluation: for three of the five datasets for which gold information is available for training (POS, IC-LABELME, and MRE) training using gold information (alone, or in conjunction with crowd information) gives better results for hard evaluation than training with crowd information only, irrespective of which measure is used. In fact, the difference between

**Table 6.2:** F1 on all the tasks using all the methods

| | | POS | PDIS | MRE | RTE | IC-LABELME | IC-CIFAR10H |
|---|---|---|---|---|---|---|---|
| 1 | Gold | 88.99 $^{17}$ | NA | 84.46 | **61.28** | **97.18** | 65.54 $^{7}$ |
| 2 | Majority Voting Silver | 76.86 $^{5}$ | 90.55 $^{5}$ | 65.24 $^{9}$ | 60.63 $^{1}$ | 79.52 $^{4}$ | 65.13 $^{7}$ |
| 3 | Dawid and Skene Silver | 76.64 $^{2}$ | 92.78 | 67.80 $^{5}$ | 60.32 $^{15}$ | 83.03 $^{9}$ | 65.53 $^{7}$ |
| 4 | MACE Silver | 77.08 $^{5}$ | 92.87 | 65.28 $^{9}$ | 60.45 $^{15}$ | 81.87 $^{6}$ | 65.40 $^{7}$ |
| 5 | Crowd Truth Silver | 78.14 $^{7}$ | 91.13 $^{6}$ | 76.11 $^{12}$ | 59.52 $^{3}$ | 83.99 $^{13}$ | 63.90 $^{4}$ |
| 6 | Sheng Repeated Labelling | 78.21 $^{7}$ | 92.00 $^{15}$ | 67.19 $^{5}$ | 58.66 $^{5}$ | 82.96 $^{9}$ | **68.36** |
| 7 | CE loss + probabilistic labels | 78.75 $^{1}$ | 92.84 | 66.44 $^{3}$ | 60.68 | 84.02 $^{13}$ | 66.43 $^{10}$ |
| 8 | KL loss + probabilistic labels | 78.92 $^{1}$ | 92.84 | 66.44 $^{3}$ | 60.43 $^{15}$ | 84.09 $^{13}$ | 66.45 $^{10}$ |
| 9 | MSE loss + probabilistic labels | 78.14 $^{7}$ | 92.88 | 66.38 $^{3}$ | 60.51 $^{15}$ | 83.61 $^{13}$ | 63.33 $^{4}$ |
| 10 | DLFC | 76.27 $^{2}$ | 92.74 | 63.87 $^{2}$ | 58.42 $^{5}$ | 83.19 $^{9}$ | **67.99** |
| 11 | MV + OA Hard Filter | 68.85 $^{10}$ | 57.56 | 64.34 $^{2}$ | 46.76 $^{12}$ | 85.37 $^{12}$ | 63.69 $^{4}$ |
| 12 | Gold + OA Hard Filter | 76.99 $^{9}$ | 65.50 | 82.38 $^{1}$ | 49.55 $^{10}$ | 94.59 $^{16}$ | 63.17 $^{15}$ |
| 13 | MV + OA Weighting | 76.86 $^{9}$ | 90.21 $^{5}$ | 65.16 $^{9}$ | 60.74 | 84.88 $^{12}$ | 65.89 $^{10}$ |
| 14 | MV + WH Weighting | NA | 90.13 $^{5}$ | 65.34 $^{9}$ | 58.53 $^{6}$ | NA | NA |
| 15 | Gold + Plank et al weighting | 89.18 $^{17}$ | 92.65 | **85.07** | 61.12 | 96.37 $^{17}$ | 64.67 $^{13}$ |
| 16 | MTLOA | 89.15 $^{17}$ | 92.82 | 84.95 | 60.99 | 96.13 $^{17}$ | 65.18 $^{7}$ |
| 17 | MTLSL | **90.06** | **92.92** | 84.87 | 61.13 | 96.82 $^{1}$ | 62.34 $^{15}$ |

**Table 6.3:** Crowd Truth Weighted F1 for all tasks using all the methods

| | | POS | PDIS | MRE | RTE | IC-LABELME | IC-CIFAR10H |
|---|---|---|---|---|---|---|---|
| 1 | Gold | 92.46 $^{17}$ | NA | 86.94 | **74.05** | **98.25** | 78.48 $^{10}$ |
| 2 | Majority Voting Silver | 85.40 $^{6}$ | 94.54 $^{5}$ | 70.02 $^{7}$ | 73.39 $^{15}$ | 87.33 $^{4}$ | 78.14 $^{7}$ |
| 3 | Dawid and Skene Silver | 85.27 $^{2}$ | 96.00 | 75.34 $^{5}$ | 73.24 $^{15}$ | 89.23 $^{9}$ | 78.50 $^{10}$ |
| 4 | MACE Silver | 85.69 $^{6}$ | 96.02 | 70.13 $^{7}$ | 73.46 $^{1}$ | 88.80 $^{6}$ | 78.43 $^{10}$ |
| 5 | Crowd Truth Silver | 86.58 $^{7}$ | 94.84 $^{6}$ | 82.17 $^{12}$ | 72.15 $^{4}$ | 90.11 $^{7}$ | 77.21 $^{4}$ |
| 6 | Sheng Repeated Labelling | 86.51 $^{7}$ | 95.43 $^{10}$ | 74.89 $^{5}$ | 71.19 $^{5}$ | 89.48 $^{9}$ | **80.56** |
| 7 | CE loss + probabilistic labels | 87.15 $^{1}$ | 96.03 | 72.80 $^{6}$ | 73.40 $^{1}$ | 90.17 $^{13}$ | 79.17 $^{10}$ |
| 8 | KL loss + probabilistic labels | 87.27 $^{1}$ | 96.01 | 73.10 $^{6}$ | 73.17 $^{16}$ | 90.20 $^{13}$ | 79.09 $^{10}$ |
| 9 | MSE loss + probabilistic labels | 86.61 $^{7}$ | 96.04 | 73.21 $^{6}$ | 73.27 $^{15}$ | 89.90 $^{13}$ | 76.74 $^{15}$ |
| 10 | DLFC | 84.76 $^{2}$ | 95.87 $^{3}$ | 66.11 $^{2}$ | 71.06 $^{5}$ | 89.57 $^{9}$ | **80.30** |
| 11 | MV + OA Hard Filter | 78.54 $^{12}$ | 67.76 $^{12}$ | 66.47 $^{2}$ | 59.20 $^{12}$ | 91.04 $^{12}$ | 77.11 $^{4}$ |
| 12 | Gold + OA Hard filter | 82.96 $^{3}$ | 74.18 $^{6}$ | 84.76 $^{1}$ | 62.12 $^{10}$ | 96.47 $^{16}$ | 76.65 $^{15}$ |
| 13 | MV + OA Weighting | 85.31 $^{6}$ | 94.26 $^{2}$ | 70.63 $^{7}$ | 73.37 $^{15}$ | 90.76 $^{12}$ | 78.80 $^{10}$ |
| 14 | MV + WH Weighting | NA | 94.29 $^{2}$ | 70.26 $^{7}$ | 71.63 $^{8}$ | NA | NA |
| 15 | Gold + Plank et al weighting | 92.60 $^{17}$ | 95.87 | 87.53 | 73.87 | 97.76 $^{17}$ | 77.76 $^{13}$ |
| 16 | MTLOA | 92.56 $^{17}$ | 95.98 | 87.42 | 73.83 | 97.60 $^{17}$ | 78.17 $^{7}$ |
| 17 | MTLSL | **93.07** | **96.05** | **87.65** | 73.66 | 98.02 $^{1}$ | 76.00 $^{15}$ |

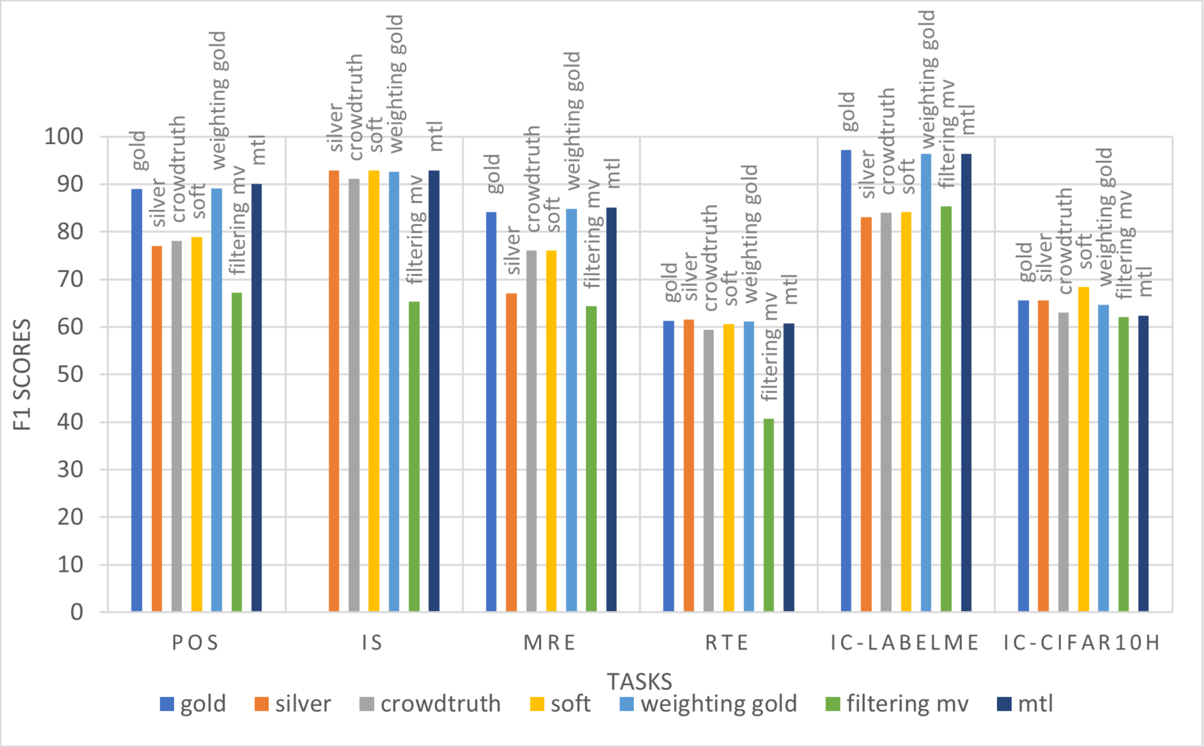


**Figure 6.2:** Graph showing the F1 scores of the best performing training approach for each category on all the datasets

the best method using gold and the best method only using crowd annotations can be quite large for these three datasets, up to 10 points in some cases (e.g., POS).

However, the answer to **RQ3** is not entirely negative, because with RTE and IC-CIFAR10H it is the other way around: with IC-CIFAR10H, the best results are obtained using crowd information only and with RTE there is no significant difference between training with gold and training using silver labels aggregated with MV. But we will already anticipate that the situation is completely reversed when soft evaluation metrics are employed; in this case, using crowd information always improves results over using gold only, as shown in Section 6.3.2.

Another finding emerging very clearly from the Tables and the Figure is that there is no evidence that the approach to using disagreement information that may appear most intuitive, filtering– using this information to remove hard items–helps with hard evaluation. With none of these datasets the best results are obtained by filtering difficult items; on the contrary, filtering typically leads to worse results, sometimes substantially so. The one exception is IC-LABELME: in this case the results obtained by filtering, while much worse than those obtained by using gold without filtering, are on par with those obtained with other ways of using crowd information.

A third observation is that the answer to **RQ4a** is mixed when using hard evaluation: leveraging crowd information in addition to gold sometimes helps, although not by much, sometimes doesn't. With two of the five datasets for which we have gold– POS and MRE– using information from the soft label to supplement gold according to the MTLSL method does improve performance over using gold labels only with all tree hard metrics; this difference is small– typically around one percentage point–but significant in the case of POS. With RTE, the best performing method depends on the metric, but the difference are never significant. In line with Card et al. [2020] who discuss statistical power in relation to dataset size, it might be worth to consider retiring this small dataset. With the two IC datasets, however, using gold only leads to significantly better results than using gold in combination with crowd information–substantially so in the case of the IC-CIFAR10H dataset. Using crowd information in addition to the hard label also helps a little bit in the IS task, when the hard label is an aggregated silver label, although the difference is not significant. Again, we must immediately point out that the situation is reversed with soft evaluation.

The answer to **RQ4b**–what is the best way to leverage crowd information in addition to gold information–is that with most datasets MTLSL is either significantly better, better, or indistinguishable from other approaches, in particular the approach by Plank *et al.*. The one exception is IC-CIFAR10H, where MTLSL performs rather poorly–but in this case the best among the approaches leveraging both gold and crowd information is the other multi-task learning approach we tested, MTLOA using observed agreement as auxiliary function.

The final observation is that the answer to **RQ5** is that there isn't a clear 'winner' among the methods not using a gold label: different methods achieve the best results depending on the task. We summarize the differences as follows, using $A >> B$ to signify that most methods of category $A$ are significantly better than most methods of category $B$; $A \sim B$ to signify that most methods of category $A$ are statistically indistinguishable from most methods of category $B$; and $A \geq B$ to signify that some methods of category $A$ are significantly better than some methods of category $B$, whereas others are equivalent.

1. On POS, the best results among the methods only using crowd information are obtained by the three methods using a soft loss function, then by using aggregation, then weighting and filtering. The performance ranking for POS shown in Figure 6.2 can be schematically summarized as follows, where $HARD_{GOLD}$ is gold training, $SOFT$ includes the Crowd Truth method, and categories are ranked by the performance of the best performing method in the category:
   $AUGMENTED_{GOLD} >> HARD_{GOLD} >> SOFT \geq FILTER_{GOLD} >> HARD_{SILVER} \geq WEIGHT_{SILVER} >> FILTER_{SILVER}$

2. On PDIS, no gold labels are available, so the silver label achieving the best results (aggregated with MACE) was used as hard label. The best results are obtained using MTLSL using this hard silver label, but augmenting hard silver with crowd information, using hard silver only, or using crowd information only with soft label methods achieve statistically indistinguishable results on this dataset. The only significant differences are between any of these methods and weighting, and between weighting and hard filtering, which gives really bad results.
   $AUGMENTED_{SILVER} \sim HARD_{SILVER} \sim SOFT >>$
   $WEIGHT_{SILVER} >> FILTER$

3. MRE is the one dataset on which different methods achieve the best results depending on which hard evaluation metric is used. Methods exploiting both gold and crowd information achieve the best results with all three hard metrics, systematically outperforming training with gold only although the difference is not significant. But among the methods not relying on gold labels, CrowdTruth aggregation obtains by far the best results in terms of F1 and especially of CT F1, with a margin of 10 points or more over other methods. Soft label methods achieve the best accuracy results, although the difference is not significant.
   $AUGMENTED_{GOLD} \sim HARD_{GOLD} >> FILTER_{GOLD} >>$
   $CT \sim_{ACCURACY} / >>_{F1} SOFT \sim_{ACCURACY} / >>_{F1} HARD_{SILVER} \geq WEIGHT_{SILVER} \geq$
   $FILTER_{SILVER}$

4. RTE is one of the datasets for which using gold information does not yield better results than using crowd information only. The results using gold information, gold augmented with crowd, silver weighing, and some of the soft loss functions are all statistically equivalent. Among the soft labelling methods, the best results are obtained by OA weighting, then soft loss using CE, then agregation. But all methods achieve roughly comparable results with all metrics, with a maximum 1-2 percentage points between the worse and the best result; again the only exception is hard filtering, that performs substantially worse.
   $AUGMENTED_{GOLD} \sim HARD_{GOLD} \sim WEIGHT_{SILVER} \sim SOFT >>$
   $HARD_{SILVER} \sim>> FILTER$

5. The best results with IC-LABELME are obtained using gold information alone (which does very slightly, but significantly, better than combining gold with crowd information). The next best results are obtained using OA for filtering or weighing silver labels–this is the only dataset in which filtering / weighing silver items proves a competitive approach. Soft labels are next, then aggregation. Using hard silver labels yields the worse results in terms of hard evaluation metrics, but this is the dataset in which probabilistic aggregation outperforms MV by the largest margin: training over the CT-aggregated labels, while not resulting in the best F1, improves performance over training with the MV labels by more than 4 points.
   $HARD_{GOLD} >> AUGMENTED_{GOLD} >> FILTER_{GOLD} >>$
   $FILTER_{SILVER} \sim WEIGH_{SILVER} >> SOFT >> HARD_{SILVER}$

6. Finally, IC-CIFAR10H is the one dataset in which using crowd information only yields significantly better results than using gold. The best results are obtained using Sheng *et al.*-style repeated labelling—the improvement is of around three points in this case—but soft-loss training also significantly outperforms gold training, which is statistically indistinguishable from silver training and from MTLOA.
   $SOFT >> WEIGH_{SILVER} >> HARD_{SILVER} \sim HARD_{GOLD} >>$
   $AUGMENTED_{GOLD} >> FILTER$

**Table 6.4:** Cross entropy between produced probabilities and and the soft labels for all tasks using all the methods. (Smaller is better)

| | | POS | PDIS | MRE | RTE | IC-LABELME | IC-CIFAR10H |
|---|---|---|---|---|---|---|---|
| 1 | Gold | 3.346 $^{16}$ | NA | 0.574 $^{17}$ | 0.771 $^{8}$ | 5.159 $^{12}$ | 2.607 $^{5}$ |
| 2 | Majority Voting Silver | 2.583 $^{4}$ | 0.397 $^{14}$ | 0.522 $^{11}$ | 0.785 $^{3}$ | 3.065 $^{4}$ | 2.627 $^{5}$ |
| 3 | Dawid and Skene Silver | 2.524 $^{4}$ | 0.300 $^{9}$ | 0.350 $^{6}$ | 0.772 $^{8}$ | 2.902 $^{10}$ | 2.554 $^{5}$ |
| 4 | MACE Silver | 2.506 $^{10}$ | 0.297 $^{9}$ | 0.460 $^{3}$ | 0.797 $^{13}$ | 2.906 $^{10}$ | 2.646 $^{5}$ |
| 5 | Crowd Truth Silver | 1.482 $^{9}$ | 0.403 $^{14}$ | 0.610 $^{16}$ | 0.673 $^{6}$ | 2.717 $^{6}$ | 1.763 $^{9}$ |
| 6 | Sheng Repeated Labelling | 1.787 $^{5}$ | 0.359 $^{9}$ | 0.310 | 0.669 | 2.572 $^{9}$ | **1.062** |
| 7 | CE loss + probabilistic labels | **1.358** | 0.273 $^{16}$ | 0.310 | 0.740 $^{9}$ | **1.638** | **1.112** |
| 8 | KL loss + probabilistic labels | **1.279** | 0.265 $^{16}$ | 0.309 | 0.742 $^{9}$ | **1.638** | **1.109** |
| 9 | MSE loss + probabilistic labels | 1.442 $^{7}$ | 0.289 $^{8}$ | **0.309** | 0.717 $^{5}$ | 1.747 $^{7}$ | 1.491 $^{8}$ |
| 10 | DLFC | 2.136 $^{6}$ | 0.275 $^{16}$ | 0.715 $^{15}$ | **0.668** | 2.798 $^{5}$ | 3.507 $^{11}$ |
| 11 | MV + OA Hard Filter | 3.243 $^{15}$ | 2.246 $^{12}$ | 0.490 $^{4}$ | 0.879 $^{14}$ | 3.684 $^{13}$ | 2.961 $^{15}$ |
| 12 | Gold + OA Hard Filter | 3.115 $^{13}$ | 1.863 $^{17}$ | 0.495 $^{4}$ | 0.879 $^{14}$ | 4.612 $^{15}$ | 2.844 $^{15}$ |
| 13 | MV + OA Weighting | 2.759 $^{2}$ | 0.372 $^{6}$ | 0.527 $^{14}$ | 0.787 $^{3}$ | 3.121 $^{2}$ | 2.615 $^{5}$ |
| 14 | MV + WH Weighting | NA | 0.379 $^{6}$ | 0.516 $^{11}$ | 0.842 $^{4}$ | NA | NA |
| 15 | Gold + Plank et al Weighting | 3.432 $^{1}$ | 0.261 $^{16}$ | 0.621 $^{16}$ | 0.779 $^{3}$ | 4.198 $^{16}$ | 2.691 $^{15}$ |
| 16 | MTLOA | 3.288 $^{12}$ | **0.245** | 0.579 $^{17}$ | 0.796 $^{13}$ | 3.926 $^{11}$ | 2.505 $^{5}$ |
| 17 | MTLSL | **1.382** | 0.618 $^{5}$ | 0.569 $^{13}$ | 0.786 $^{3}$ | **1.642** | 4.032 $^{1}$ |

We further analyse these results on a task-by-task basis in Section 6.4, with the aim to explain these dataset-dependent differences.

One final consideration: it can be observed that the three evaluation metrics tend to be aligned, in the sense that the methods performing best on a given task are the same irrespective of the evaluation used, with the few exceptions noted.

### 6.3.2 Evaluation against soft labels

Given the empirical evidence challenging the assumption that it is always possible to assign a unique label to items in cognitive tasks reviewed in Chapter 2, the form of evaluation discussed in the previous subsection –testing models against gold labels – while standard in NLP and in AI, does not tell the complete story. In this dissertation, therefore, we also evaluate current methods for training with disagreement using the 'soft' evaluation metrics discussed in Section 3.3. The results are shown in Tables 6.4 to 6.7.

Arguably, the main result of this work is that the answer to **RQ3**, which as seen in Section 6.3.1 is mainly negative when using hard evaluation metrics, becomes positive with soft evaluation metrics: i.e., the ranking among methods for learning from disagreement seen in the previous Section is to a large extent reversed when these methods are evaluated using a soft evaluation metric, so that methods *not* using gold information generally outperform hard-training methods for all tasks and all metrics.

The answer to **RQ5**–which of these methods performs best–again depends on the task and, to a lesser extent, on the metric, but for almost all metrics and almost all tasks the best results are obtained by some form of soft loss trainin g or repeated labelling.

**Table 6.5:** Jensen-Shannon Divergence results on all tasks using all the methods. (Smaller is better.)

| | | POS | PDIS | MRE | RTE | IC-LABELME | IC-CIFAR10H |
|---|---|---|---|---|---|---|---|
| 1 | Gold | 0.413 $^{11}$ | NA | 0.251 $^{12}$ | 0.415 $^{7}$ | 0.547 $^{12}$ | 0.405 $^{10}$ |
| 2 | Majority Voting Silver | 0.353 $^{4}$ | 0.218 $^{16}$ | 0.166 $^{4}$ | 0.416 $^{7}$ | 0.452 $^{4}$ | 0.399 $^{10}$ |
| 3 | Dawid and Skene Silver | 0.353 $^{4}$ | 0.129 $^{17}$ | 0.156 $^{7}$ | 0.416 $^{7}$ | 0.449 $^{5}$ | 0.404 $^{10}$ |
| 4 | MACE Silver | 0.351 $^{10}$ | 0.129 $^{17}$ | 0.163 $^{3}$ | 0.415 $^{7}$ | 0.448 $^{10}$ | 0.395 $^{10}$ |
| 5 | Crowd Truth Silver | 0.236 $^{7}$ | 0.243 $^{13}$ | 0.297 $^{15}$ | 0.425 $^{12}$ | 0.428 $^{6}$ | 0.417 $^{1}$ |
| 6 | Sheng Repeated Labelling | 0.318 $^{9}$ | 0.268 $^{10}$ | **0.136** | 0.426 $^{11}$ | 0.417 $^{9}$ | 0.415 $^{1}$ |
| 7 | CE loss + probabilistic labels | **0.207** | 0.146 $^{9}$ | 0.148 $^{6}$ | 0.413 $^{13}$ | **0.201** | 0.427 $^{1}$ |
| 8 | KL loss + probabilistic labels | **0.206** | 0.150 $^{7}$ | 0.148 $^{6}$ | 0.413 $^{13}$ | **0.201** | 0.428 $^{1}$ |
| 9 | MSE loss + probabilistic labels | 0.280 $^{17}$ | 0.128 $^{17}$ | 0.148 $^{6}$ | 0.416 $^{7}$ | 0.208 $^{7}$ | 0.431 $^{1}$ |
| 10 | DLFC | 0.342 $^{6}$ | 0.220 $^{2}$ | 0.177 $^{17}$ | 0.426 $^{11}$ | 0.430 $^{6}$ | **0.368** |
| 11 | MV + OA Hard Filter | 0.397 $^{13}$ | 0.351 $^{12}$ | 0.169 $^{14}$ | 0.430 $^{12}$ | 0.489 $^{13}$ | 0.407 $^{2}$ |
| 12 | Gold + OA Hard Filter | 0.426 $^{15}$ | 0.303 $^{5}$ | 0.241 $^{10}$ | 0.421 $^{1}$ | 0.539 $^{15}$ | 0.412 $^{2}$ |
| 13 | MV + OA Weighting | 0.353 $^{4}$ | 0.232 $^{10}$ | 0.166 $^{4}$ | 0.410 $^{17}$ | 0.464 $^{2}$ | 0.392 $^{10}$ |
| 14 | MV + WH Weighting | NA | 0.256 $^{10}$ | 0.167 $^{4}$ | 0.422 $^{16}$ | NA | NA |
| 15 | Gold + Plank et al Weighting | 0.415 $^{11}$ | 0.163 $^{8}$ | 0.258 $^{1}$ | 0.417 $^{1}$ | 0.537 $^{16}$ | 0.407 $^{2}$ |
| 16 | MTLOA | 0.413 $^{11}$ | 0.178 $^{15}$ | 0.250 $^{12}$ | 0.418 $^{1}$ | 0.531 $^{11}$ | 0.401 $^{10}$ |
| 17 | MTLSL | 0.236 $^{7}$ | **0.096** | 0.172 $^{11}$ | **0.404** | **0.201** | 0.415 $^{1}$ |

**Table 6.6:** Cosine Similarity between the entropy of the produced distribution and the annotation label distribution for all tasks using all the methods

| | | POS | PDIS | MRE | RTE | IC-LABELME | IC-CIFAR10H |
|---|---|---|---|---|---|---|---|
| 1 | Gold | 0.659 $^{11}$ | NA | 0.655 $^{12}$ | 0.567 $^{3}$ | 0.551 $^{16}$ | 0.389 $^{5}$ |
| 2 | Majority Voting Silver | 0.758 $^{10}$ | 0.115 $^{13}$ | 0.478 $^{4}$ | 0.570 $^{7}$ | 0.778 $^{3}$ | 0.383 $^{5}$ |
| 3 | Dawid and Skene Silver | 0.762 $^{10}$ | 0.176 $^{9}$ | 0.700 $^{7}$ | 0.571 $^{7}$ | 0.797 $^{10}$ | 0.391 $^{5}$ |
| 4 | MACE Silver | 0.750 $^{10}$ | 0.183 $^{9}$ | 0.548 $^{15}$ | 0.560 $^{2}$ | 0.777 $^{3}$ | 0.379 $^{5}$ |
| 5 | Crowd Truth Silver | 0.885 $^{7}$ | 0.116 $^{13}$ | 0.717 $^{7}$ | 0.589 $^{6}$ | 0.840 $^{10}$ | 0.472 $^{9}$ |
| 6 | Sheng Repeated Labelling | 0.873 $^{7}$ | 0.167 $^{4}$ | **0.772** | **0.590** | 0.860 $^{17}$ | 0.546 |
| 7 | CE loss + probabilistic labels | **0.899** | 0.204 $^{15}$ | 0.761 $^{6}$ | 0.579 $^{9}$ | **0.979** | 0.546 |
| 8 | KL loss + probabilistic labels | **0.907** | 0.211 $^{15}$ | 0.763 $^{6}$ | 0.579 $^{9}$ | 0.978 $^{7}$ | **0.547** |
| 9 | MSE loss + probabilistic labels | 0.888 $^{7}$ | 0.191 $^{7}$ | 0.761 $^{6}$ | 0.584 $^{5}$ | 0.978 $^{7}$ | 0.506 $^{6}$ |
| 10 | DLFC | 0.849 $^{6}$ | 0.207 $^{15}$ | 0.688 $^{3}$ | 0.589 $^{6}$ | 0.852 $^{6}$ | 0.331 $^{13}$ |
| 11 | MV + OA Hard Filter | 0.698 $^{13}$ | 0.065 $^{5}$ | 0.590 $^{15}$ | 0.517 $^{4}$ | 0.697 $^{14}$ | 0.390 $^{5}$ |
| 12 | Gold + OA Hard Filter | 0.729 $^{4}$ | 0.071 $^{5}$ | 0.678 $^{5}$ | 0.518 $^{4}$ | 0.592 $^{12}$ | 0.379 $^{5}$ |
| 13 | MV + OA Weighting | 0.720 $^{4}$ | 0.136 $^{14}$ | 0.455 $^{2}$ | 0.570 $^{7}$ | 0.755 $^{4}$ | 0.372 $^{5}$ |
| 14 | MV + WH Weighting | NA | 0.161 $^{4}$ | 0.501 $^{4}$ | 0.571 $^{7}$ | NA | NA |
| 15 | Gold + Plank et al Weighting | 0.650 $^{16}$ | 0.241 $^{16}$ | 0.641 $^{1}$ | 0.570 $^{7}$ | 0.597 $^{16}$ | 0.391 $^{5}$ |
| 16 | MTLOA | 0.667 $^{11}$ | **0.264** | 0.655 $^{12}$ | 0.567 $^{3}$ | 0.625 $^{11}$ | 0.393 $^{5}$ |
| 17 | MTLSL | 0.876 $^{5}$ | 0.071 $^{5}$ | 0.433 $^{2}$ | 0.563 $^{1}$ | 0.976 $^{7}$ | 0.352 $^{13}$ |

**Table 6.7:** Pearson correlation between the entropy of the produced distribution and the annotation label distribution for all tasks using all the methods

| | | POS | PDIS | MRE | RTE | IC-LABELME | IC-CIFAR10H |
|---|---|---|---|---|---|---|---|
| 1 | Gold | 0.399 $^{11}$ | NA | 0.223 $^{4}$ | 0.037 $^{7}$ | -0.016 $^{2}$ | 0.127 $^{5}$ |
| 2 | Majority Voting Silver | 0.517 $^{10}$ | -0.104 $^{11}$ | 0.214 $^{4}$ | 0.043 $^{7}$ | 0.026 $^{11}$ | 0.118 $^{5}$ |
| 3 | Dawid and Skene Silver | 0.504 $^{10}$ | 0.029 $^{9}$ | 0.382 $^{7}$ | 0.039 $^{7}$ | 0.111 $^{11}$ | 0.125 $^{5}$ |
| 4 | MACE Silver | 0.513 $^{10}$ | 0.032 $^{9}$ | 0.265 $^{3}$ | 0.022 $^{1}$ | 0.139 $^{11}$ | 0.112 $^{5}$ |
| 5 | Crowd Truth Silver | 0.642 $^{7}$ | -0.113 $^{2}$ | 0.293 $^{7}$ | 0.037 $^{7}$ | 0.194 $^{10}$ | 0.160 $^{9}$ |
| 6 | Sheng Repeated Labelling | 0.635 $^{7}$ | -0.098 $^{10}$ | **0.511** | 0.030 $^{2}$ | 0.284 $^{8}$ | 0.217 |
| 7 | CE loss + probabilistic labels | **0.656** | 0.051 $^{15}$ | 0.444 $^{6}$ | 0.056 | 0.407 $^{9}$ | 0.215 |
| 8 | KL loss + probabilistic labels | **0.663** | 0.053 $^{15}$ | 0.450 $^{6}$ | 0.059 | 0.403 $^{9}$ | **0.217** |
| 9 | MSE loss + probabilistic labels | 0.640 $^{7}$ | 0.047 $^{15}$ | 0.435 $^{6}$ | 0.058 | **0.425** | 0.190 $^{6}$ |
| 10 | DLFC | 0.603 $^{6}$ | -0.040 $^{17}$ | 0.010 $^{17}$ | 0.025 $^{3}$ | 0.263 $^{6}$ | 0.119 $^{5}$ |
| 11 | MV + OA Hard Filter | 0.411 $^{12}$ | -0.023 $^{17}$ | 0.208 $^{4}$ | -0.061 $^{12}$ | 0.192 $^{10}$ | 0.121 $^{5}$ |
| 12 | Gold + OA Hard Filter | 0.451 $^{6}$ | -0.029 $^{17}$ | 0.227 $^{3}$ | -0.046 $^{4}$ | 0.130 $^{11}$ | 0.107 $^{5}$ |
| 13 | MV + OA Weighting | 0.517 $^{10}$ | -0.102 $^{10}$ | 0.211 $^{4}$ | 0.056 | 0.195 $^{10}$ | 0.100 $^{5}$ |
| 14 | MV + WH Weighting | NA | -0.091 $^{10}$ | 0.217 $^{4}$ | **0.065** | NA | NA |
| 15 | Gold + Plank et al Weighting | 0.395 $^{11}$ | 0.067 | 0.217 $^{4}$ | 0.030 $^{3}$ | -0.020 $^{2}$ | 0.129 $^{5}$ |
| 16 | MTLOA | 0.408 $^{12}$ | **0.079** | 0.215 $^{4}$ | 0.035 $^{7}$ | -0.016 $^{2}$ | 0.124 $^{5}$ |
| 17 | MTLSL | 0.612 $^{6}$ | 0.020 $^{7}$ | 0.120 $^{13}$ | 0.047 $^{7}$ | 0.376 $^{7}$ | 0.105 $^{5}$ |

The answer to **RQ3** is also uniformly positive: using crowd information always helps improving the results over training using gold only, for all evaluation metrics and all datasets. In answer to **RQ4**, some form of Multi-Task Learning with an auxiliary function capturing disagreements is usually the best approach, with MTLSL in particular achieving pretty good results in many cases.

- Almost all training methods using gold information, except for MTLSL, achieve significantly worse performance under all soft metrics with POS, as with all other datasets. Soft-loss training methods perform best, with CE and KL loss training performing significantly better than all other methods according to all soft evaluation metrics. Interestingly, MTLSL performs better than most soft-labelling methods, whereas MTLOA doesn't. Also worth noting that CrowdTruth aggregation achieves better results than the other aggregation methods. This can be loosely summarized as follows:
  $SOFT_{CE,KL} \sim_{CE} / >>_{JSD,CS} MTLSL \geq CT \sim SOFT \geq$
  $HARD_{SILVER} >> WEIGHT \sim FILTER >> HARD_{GOLD}$

- With PDIS, MTL methods (using silver as the hard label) perform best according to all four soft evaluation metrics, but the type of MTL that works best depends on the evaluation, and sometimes the difference in results is quite substantial. E.g., with cross entropy, MTLOA is the best type of training, but MTLSL is the worst. Soft loss methods are next best, then methods that rely on a prior aggregation. It should be noted that the best results with soft-loss functions with this dataset are obtained using the posterior of probabilistic aggregation methods as target. It should also be noted that all methods do pretty badly at predicting the entropy of the annotator labels distribution with this dataset, whether computed using

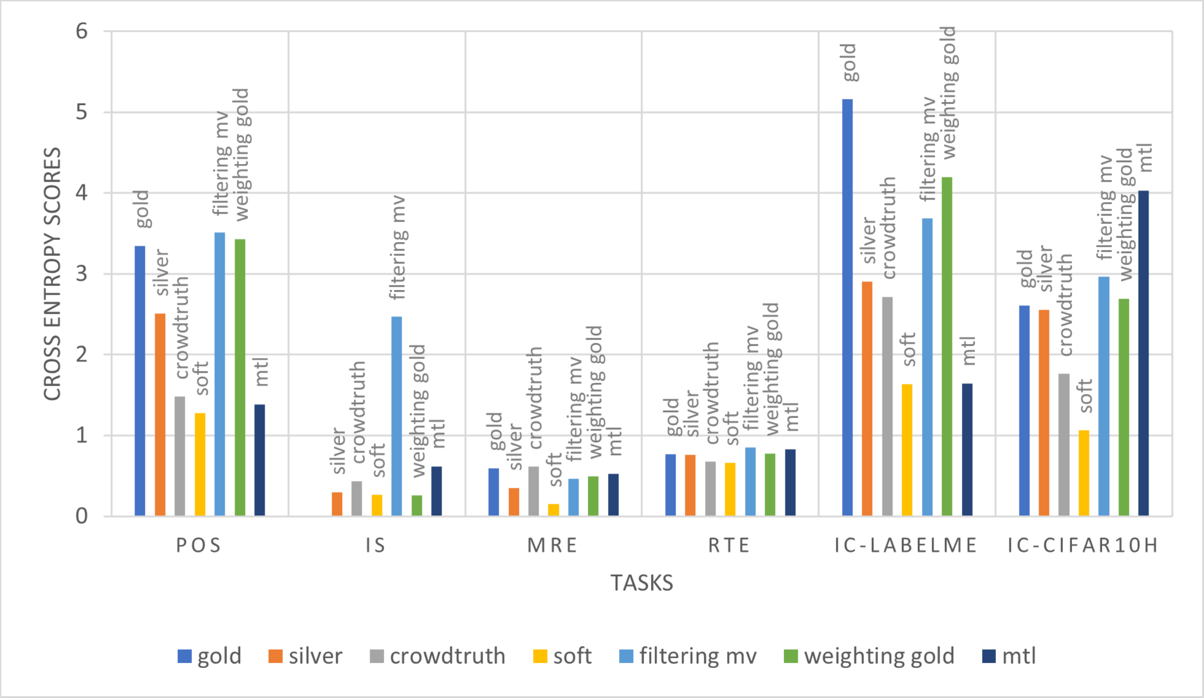


**Figure 6.3:** Graph showing the Cross Entropy scores of the best performing training approach for each category on all the datasets (lower is better)

cosine similarity or, even worse, using Pearson correlation.
$MTLOA/MTLSL >> SOFT >> HARD_{SILVER} \geq$
$WEIGHT \geq FILTER$.

- With MRE, soft-labelling methods perform best, but different types of training achieve the best results depending on the evaluation used. Repeated labelling generally performs best, followed by soft-loss methods, except for cross-entropy where it is the other way around; but the difference is typically not significant.
$SHENG >> SOFTLOSS \geq HARD_{SILVER} \geq FILTER, WEIGHT \geq$
$AUGMENTED_{GOLD} \geq HARD_{GOLD}$.

- One striking aspect of the results with RTE is that the results of the different methods are much closer than with other datasets, although significant differences do emerge. In particular, although the methods relying only on crowd information outperform gold training and training on aggregated silver labels according to most soft metrics, the differences are much smaller, and MTLSL outperforms the soft-labelling methods in terms of JSD. For this dataset, Sheng *et al.*'s repeated labelling and DLC achieve the best results in terms of cross-entropy (there is a small difference, but it is not significant) and cosine similarity; the difference from other soft-loss methods is significant. Another noticeable result is that the Pearson correlation between the entropy of the produced distribution and that of the target distribution is mostly near 0. The results with this dataset are difficult to summarize because the soft metrics do not all point to the same ranking, but as a first approximation, we can say that:

$SOFT_{SHENG,DLFC} \geq CT \geq MTLSL \geq HARD_{SILVER} \geq$
$WEIGHT, FILTER \geq HARD_{GOLD}$.

- With IC-LABELME, the best performing method on all metrics is training using a soft loss function with softmax distribution, although again MTLSL is a very close second best in most cases, and is equivalent to soft-loss training when evaluated using Cross Entropy or JSD. Unsurprisingly perhaps, for this task using gold only for training results in a really bad match regarding predicted entropy.
  $SOFT \geq MTLSL >> CT \geq HARD_{SILVER} >>$
  $WEIGHT >> FILTER \geq HARD_{GOLD}$.

- And finally, for IC-CIFAR10H soft-labelling methods perform clearly better although again which method performs best–soft-loss, repeated labelling - depends on the measure used.
  $SOFT >> CT, HARD_{SILVER} \geq FILTER, WEIGHT \geq HARD_{GOLD}$.

## 6.4 A Dataset-by-Dataset Analysis of the Results

We just saw in Section 6.3 that the relative performance of current methods for learning from disagreement varies greatly from dataset to dataset, both with hard and with soft evaluation metrics. The aim of this Section is to analyse in greater depth these differences, looking at each dataset in isolation, aiming to explain how the pattern of results observed in that dataset relate to its characteristics.

Each subsection includes sections devoted to the results obtained on a dataset by training with gold (with or without supplemented with crowd information), with aggregated labels, and with soft labels only.

### 6.4.1 Part-of-Speech Tagging

The key characteristics of the POS dataset (see Tables 2.1 and 2.2) are that it has the second highest number of items (14,439), average coder accuracy is high (.93), and the mean number of annotations per item is also fairly high (16.37). However, observed agreement is relatively low (.73),and the quality of aggregated labels is low as well. Finally, while the raw annotator entropy is fairly low (.13), the Best Distribution Entropy (BDE) is fairly high (.39).

**Gold vs. non-gold**

As discussed in Section 6.3, with the POS dataset we see clear differences in performance between the models using gold labels and those using silver or soft labels, in both directions. Methods using gold labels do clearly better in terms of hard evaluation metrics, although using soft information helps; whereas methods only using soft labels do clearly better in terms of soft evaluation metrics.

This substantial difference between using and not using gold is surprising given the high coder accuracy and the substantial number of annotations per item, and the well-known findings of, e.g., [Snow et al., 2008, Sheng et al., 2008] that the quality of labels produced by the crowd is comparable to that of labels produced by experts provided that sufficient coders of sufficient quality are employed. To understand this surprising result, we carried out a more in-depth analysis of the data, whose results are shown in Table 2.1 and Table 6.8. First of all, the high mean number of annotators per item is deceptive: whereas several items have a high number of annotations (177), for others the number of annotations is much lower, so that the median of this figure is only 5. Second, Table 6.8 shows that this dataset is not uniform, but can be partitioned in two subsets with very different characteristics. 80% of the judgments in the dataset are about nouns, even though the constitute only 27.7% of the total number of items. The coder accuracy for these items is very high, almost 98%, and so is the average number of annotations per item. By contrast, only 20% of the judgments are about the remaining 72% of items, and coder accuracy on these is much lower. This suggests that the quality of crowd information for the great majority of items is not high enough. As we will see, Snow *et al.*'s hypothesis does hold with datasets with uniformly high coder accuracy and number of annotations per item.

**Table 6.8:** Nouns vs Non-Nouns in the POS dataset

| | Nouns | Others |
|---|---|---|
| Percentage of items in the subset | 27.72 | 72.28 |
| Percentage of judgments in category per | 80.13 | 19.87 |
| Average number of annotations per item | 12.57 | 3.80 |
| Average annotator accuracy | 97.89 | 69.08 |
| Average item observed agreement | 0.804 | 0.695 |
| MV aggregated label accuracy | 85.94 | 77.52 |

### Using soft Labels to supplement gold labels

In Chapter 5, we saw that for this dataset using crowd information in addition to gold improves upon using gold alone when the models are evaluated using one metric from each evaluation paradigm (see Chapter 3.3); accuracy, and cross entropy entropy correlation. This finding also holds for the other evaluation metrics used in this research. MTLSL stands out as the best method for learning hard or weighted truth for the POS dataset under all three hard evaluation metrics. This method, which targets the soft label distribution as an auxiliary task to supplement the gold labels, achieves +1.03, +1.06, and +0.61 significant points over training over gold alone when evaluated using accuracy, F1 and CT F1 respectively. The other methods that augment gold labels with information from the crowd, MTLOA and [Plank et al., 2014a], also outperform the gold, although not by a significant margin.

The fact that the training methods leveraging crowd information improve over gold training suggest that the crowd provides information that usefully supplements the gold labels. As already mentioned, the POS dataset is characterized by a combination of relatively high number of judgments per item, accurate coders, relatively low observed agreement between them, and moderate 'Best Distribution Entropy'. It would seem then plausible that it is the quality, quantity *and diversity* of crowd judgments that leads to the crowd information improving over gold– which in turn suggests that the low agreement is indeed due to the fact that more than one interpretation is possible for several items in this dataset [Plank et al., 2014b]. As we will see, this hypothesis that these are the conditions under which using soft labels in addition to gold labels improves performance holds for the other datasets as well.

It is less surprising that MTLSL also outperforms gold training according to all four soft evaluation metrics: it produces a distribution less divergent from the annotators' distribution (as measured using cross entropy and Jensen-Shannon Divergence) and better captures item confusion (as measured by Cosine similarity and Pearson correlation). Of the other approaches to using crowd information to supplement gold labels (MTLOA and Plank et al. [2014a]), MTLOA always outperforms gold training, sometimes significantly so, but Plank et al. [2014a] falls behind gold training, significantly so when evaluated using cross entropy and cosine similarity.

The fact that the MTLSL method for supplementing gold labels is more effective than MTLOA and the Plank et al. [2014a] method suggests that targeting the distribution of labels is more useful than targeting observed agreement or confusion among labels with this dataset.[6]

### Learning from aggregated labels

With hard evaluation, training using aggregated labels produces results significantly worse than training from gold labels with this dataset, and slightly worse than training from soft labels. With soft evaluation, training with aggregated labels gets worse or significantly worse results than training with soft labels, but better or substantially better than training with gold labels. Majority Voting (MV), Dawid and Skene (D&S) and MACE achieve comparable results according to all metrics.

As we will see, this result is not unusual: training against the entire 'soft label' yields as good or better results than training against aggregated silver labels with all of our datasets. This suggests that the distribution of labels produced by the annotators provides useful information, which is lost when the label is aggregated. Training with aggregated labels only matches training with soft labels with datasets such as PDIS, where average coder accuracy is relatively low, yet there is an abundance of annotations per item, allowing an aggregation method to learn accurate models, reflected in a high quality of the aggregated labels, much higher than obtained with MV. With POS, aggregation methods do not outperform MV on this task for most of the evaluation metrics, and also aggregation methods perform about the same wrt hard

[6] It should however be noted that in Plank et al. [2014a] where the label confusion is computing using the annotations of *two* expert annotators, the Plank *et al.* method outperformed gold training. Perhaps, a better way exists to extend the Plank et al. [2014a] method to a multi-annotator scenario than the one used here.

evaluation as the dlc method [Rodrigues and Pereira, 2018], which also attempts to learn models of the coders and uses them to weigh interpretations.

### Learning from soft labels

The best results for this dataset without using gold labels for training are obtained by two soft-loss methods using as target the distribution obtained by softmaxing the raw proportions, rather than standard normalization or the output of probabilistic aggregation. The best results are obtained by using kl as a soft loss function; training using ce as a soft loss function achieves slightly worse results but but not significantly different. These two methods also obtain the best results according to soft evaluation metrics, both in producing a probability distribution least divergent from the annotators' label distribution, and for capturing confusion as measured by entropy. These results provide further evidence that the crowd information included with this dataset provides useful information for learning.

The next best results are obtained by a cluster of methods including mse soft-loss, the Repeated labelling method by Sheng *et al.*, and the soft approximate of the Crowd Truth aggregation method (see section 6.2.1. These methods are significantly outperformed by the kl and ce soft-loss methods, but significantly outperform the other soft-label method, dlc. All soft labelling methods outperform training methods using aggregated labels.

### Filtering and weighting by inverse difficulty

Using crowd information (specifically, Observed Agreement on an item) to filter 'hard' items generally results in significantly worse hard evaluation performance: training with gold data with hard items filtered always results in worse performance than training with all the gold data, and training with mv silver labels with hard items filtered always results in worse performance than training with the entire dataset with one exception discussed below, ic-labelme. pos is a clear illustration of this finding. For f1 evaluation for example, filtering then training on gold labels falls 12 f1 points below Gold training without pre-filtering, and training on labels aggregated using Majority Voting, then filtered falls 8 f1 points below training using mv Silver without pre-filtering. With soft evaluation, however– with which hard label methods generally perform worse anyway– the effect of filtering is less clear-cut: in some cases we see an improvement, in other ones we don't. With pos, training on gold labels after filtering hard items (Gold + oa Filter) leads to significantly better soft evaluation results than training on Gold labels without pre-filtering; however, the reverse is the case for mv + oa Filter and mv training.

Augmenting silver (mv) labels by weighting the loss of each item according to some measure of confusion, such as the observed agreement for that item (oa weighting) generally works better than filtering in terms of hard evaluation: besides again yielding much better results with ic-labelme, it also achieves better results than training with unweighted MV labels on MRE, RTE and IC-CIFAR10H, although the difference is generally not significant. But typically weighing doesn't affect soft evaluation results. With POS we do not see an improvement over using MV labels alone according to either hard or soft metrics; in fact we find significantly worse cosine similarity and cross entropy results. This may suggest that given the complexity of the annotations as discussed above, OA alone is not informative about the nature of disagreements for this task POS. (Note that the probabilistic model of item difficulty annotation proposed by Whitehill et al. [2009] is not applicable to this dataset as it only applies to binary tasks.)

### 6.4.2 Information Status Classification

In the PDIS dataset, gold labels are only available for testing, not for training, so it is not possible to report results for training with gold, and our discussion will focus on the results obtained with aggregated labels and soft labels. We did evaluate the performance on this dataset of 'hybrid' methods relying both on a hard label and on information from soft labels (Plank *et al.*, MTLSL), but using as the hard label the most accurate aggregated label (obtained with MACE, but the same accuracy is obtained using D&S), instead of a gold label as in other datasets. So the results with 'gold' and 'augmented gold' are not directly comparable to those obtained with other datasets.

The key characteristics of PDIS (see Tables 2.1 and 2.2) are that it has the highest number of items (96,305), and the average number of annotations per item is also fairly high (11.87). Observed agreement is medium high (.81). However, average coder accuracy is mediocre (.78), and the percentage of 'expert' coders is low (.71). Notwithstanding this, the quality of aggregated labels is high, .98. Finally, the entropy statistics are the opposite as with POS: while the raw annotator entropy is fairly high (.38), the Best Distribution Entropy (BDE) is one of the lowest (.09).

#### Using soft labels to supplement (silver) hard labels

As with POS, MTLSL is the best method for learning hard or weighted truth for the PDIS under all three hard evaluation metrics; unlike with POS, however, the improvement is not significant. MTLOA and [Plank et al., 2014a] also perform on par with MACE. We would argue that the explanation proposed when discussing POS in the previous subsection–that soft labels provide information that can lead to improvements over the hard labels when the dataset contains sufficient quality, quantity, and diversity in the soft labels–explains the results with PDIS as well. In the case of PDIS, while we have a high number of crowd judgments, their quality is lower than with POS (average accuracy .78, percentage of 'expert' coders .71) and above all we have much less diversity, as measured by our Best Distribution Entropy measure: .09, as opposed to .39 with POS.

With soft evaluation, augmenting MACE with crowd information generally improves results. MTLOA and the [Plank et al., 2014a] method outperform MACE and in fact all the aggregated and soft labelling methods according to three out of the four evaluation metrics (all except Jensen-Shannon Divergence). However, MACE outperforms MTLSL using all evaluation metrics except Jensen-Shannon Divergence. This suggests that information about label and/or annotator confusion are more useful for this dataset than the probabilistic output of the MACE aggregation model used as soft label for this dataset (see section 4.5.1).

### Learning from aggregated labels

As shown in Table 2.1, PDIS is a very mixed dataset in terms of annotator performance. Average annotator accuracy is not very high, 78%, and the variation is much wider than in the datasets. In addition, the annotators did very varying amounts of work, annotating from about 1% to 13% of the dataset; but the majority of the annotations was produced by the annotators doing the most work. We would therefore expect, first of all, probabilistic aggregation methods to perform much better than Majority Voting wrt hard evaluation, as MV's assumption that all annotators have similar ability clearly doesn't hold, while probabilistic aggregation methods have enough evidence to learn accurate characterizations of the annotators that produced most of the labels, unlike with POS. This prediction is borne out, first, by the fact that the quality of probabilistically aggregated labels (98%) is much higher than the quality of MV labels (89%) (see Table 2.2) and the quality of aggregated labels in POS (at most 80%). And second, by the fact that training with probabilistically aggregated labels outscores training with MV labels by at least 2 percentage points with all three hard evaluation metrics (see results in Tables 6.1, 6.2, and 6.3).

A second expectation is that having access to the entire distribution of labels produced by the crowd should be less informative in terms of predicting the most likely label than in the case of POS, primarily because the diversity of labels as estimated by the Best Distribution Entropy is so low (.09) but also because the quality of coders as estimated in terms of observed agreement is much lower and we also have high variance in coder ability. (One could also think that given that probabilistic aggregation methods achieve such high accuracy it would be difficult to improve upon it, but this is not the case e.g., with IC-CIFAR10H, as we will see.) And indeed, for this dataset, training using aggregated labels performs on par with training using soft labelling methods.

### Learning from soft labels

The soft label does not appear to be entirely uninformative, however. MTLSL, making use of both the hard aggregated silver and the soft label, still marginally outperforms training with aggregated silver only. Also, and as importantly, soft-labelling methods and/or versions of MTL outperform pure aggregated label training with all soft evaluation metrics.

While using aggregated labels performs on par with augmented methods and most of the soft labelling methods (the CE/KL/MSE soft loss methods) with hard metrics, they are outperformed by these soft-loss methods when it comes to learning the entropy of the annotator-produced distributions, a proxy for labelling uncertainty that we measured using cosine similarity and the Pearson correlation of the entropy.

However, it is striking that all methods are quite bad at predicting entropy with this dataset. This is also the only dataset for which the best distribution was obtained from the posterior of a probabilistic aggregation method (MACE), rather than directly from raw annotations using softmax or standard normalization. The likely explanation for these two findings is that crowd information for this dataset is very noisy, so no method can learn to predict them accurately. The second finding also likely explains why none of the soft loss methods (CE/KL/MSE) improve over their hard label counterpart for this dataset, even insignificantly: the soft labels obtained via MACE have had too much disagreement information removed to be useful.

As mentioned above, the CE/KL/MSE soft loss methods are on par with the aggregated (and augmented) methods when it comes to hard evaluation metrics. These methods are also on par with the DLC method when evaluating using Accuracy and F1 and slightly outperform it when evaluating using CT F1. Like the MACE D&S aggregation models, DLC learns ground truth by learning annotator reliability. But the CE/KL/MSE soft loss methods and the DLC method all significantly outperform the Sheng Repeated Labelling method, which is based on raw coder annotations. This again shows that for PDIS, gains in hard evaluation performance are seen with models that discriminate between annotators/annotations. Further evidence is the fact that although the CT soft aggregation method outperforms MV, it is the least performing soft label method when it comes to hard evaluation. While the best of the CE/KL/MSE always outperforms the best aggregated method when evaluated using soft metrics, the same cannot be said of DLC, Sheng Repeated Labelling and CT soft aggregation method, as we will see when discussing other datasets.

### Filtering and weighting by inverse difficulty

As with POS, pre-filtering then training resulted in lower performance then training on MV labels alone when evaluated using all the metrics. While we did not have Gold labels for training, we observed that pre-filtering and training using the very high quality MACE aggregated labels also leads to a worsened performance using all the metrics.

While weighting using Observed Agreement or using the item difficulty scores produced by the Whitehill et al. [2009] method outperformed training using MV alone using soft metrics, it did not lead to a higher hard evaluation performance. One interesting finding is that weighing with probabilistically inferred inverse difficulty generally results in worse performance than weighing with OA.

### 6.4.3 (Medical) Relation Extraction

The key characteristics of MRE (see Tables 2.1 and 2.2) are that it is one of the smallest datasets we studied, with only 975 items, and coder accuracy is also pretty low– average coder accuracy is even lower than with PDIS (.76), and the percentage of 'expert' coders is much lower (.58), although observed agreement among annotators is reasonably high (.86). The average number of annotations per item is fairly high (15.3), but the quality of aggregated labels is the lowest among all of our datasets (highest is .77). Both raw annotator entropy and Best Distribution Entropy are fairly high, .31.

Another interesting observation about the dataset is that, going by the gold label, the dataset is very imbalanced, with a ratio of 2.94:1 between class 0 ($false$) and class 1 ($true$).As a result, Accuracy ranking often differs from F1 or CT F1 ranking. Because we use the class weighted version of the F1 metric, it is expected that the results will differ, as the metric will assign a higher score to the model that produces more correct answers of class 0. And, on this note, a striking result is that if the goal was to learn the majority class (i.e. evaluated using F1 or the weighted F1 metric), the Crowd Truth method outperforms all other methods for learning from crowds, confirming the results obtained by Dumitrache et al. [2018b].

#### Gold vs. non-gold

With MRE, as with POS, we find a large difference in performance between the results obtained training from the crowd only and training using gold, when measured using hard evaluation metrics. We observed a similar difference with POS (Section 6.4.1), and pointed out that the most likely explanation was poor annotator accuracy for nearly all of the classes except Nouns and Pronouns. The same explanation applies to MRE, and we can see this without digging in the dataset: MRE (and IC-LABELME, discussed in Section 6.4.5) are the datasets with the lowest average annotator accuracy and the lowest proportion of 'good' annotators when assessed against gold. In other words, these are the datasets where the crowd produced labels least like the gold. It is therefore not surprising that the models trained on these labels also produce labels that substantially differ from the gold labels.

#### Using Soft Labels to Supplement Gold labels

As noted above, using gold in training yields the best results for MRE with hard evaluation. The best results with all hard metrics are obtained by supplementing gold with soft labels, but the improvement over using gold only is typically not significant. (MTLSL works slightly better according to Accuracy and CT F1, Plank *et al.* according to F1.) The fact that we see a small but not significant improvement is, we believe, consistent with the hypothesis proposed in Section 6.4.1 about the conditions under which this happens: MRE has a fairly high BDE, indicative of a good level of diversity, but not as high as that of POS; it has a good number of annotations per item; but the size of the dataset is likely too small to observe an effect, and coder accuracy is also fairly low.

With soft evaluation, we find that one of the gold + soft label methods achieves slightly better results than training with gold only according to CE or JSD, but slightly worse when measured using cosine similarity and entropy correlation. The most likely explanation is that in the MRE dataset the annotators have a low average accuracy, so the entropy may not be predictable; but it may be again a matter of size.

### Learning from aggregated labels

Several interesting observations can be made from the hard evaluation tables in Section 6.3.1. First of all, we can see that unlike with POS, where soft labelling training generally outperformed training with aggregated labels, pretty much all crowd-only training methods achieve about the same results in terms of Accuracy (Table 6.1), although some interesting differences can be seen with F1 and CT F1. Second, we find that again the comparison between probabilistic agggregation methods and MV is very much affected by the hard evaluation metrics. With Accuracy, all aggregation methods perform about the same. With F1 and CT F1, however, D&S performs much better than both MV and MACE both in terms of hard and of soft evaluation–this is the only dataset where we find a substantial difference between D&S and MACE under either form of evaluation. Together with the finding about the performance of CT aggregation, this result suggests that D&S is better than either MV or MACE at modelling the main class.

Third, we find that the one crowd-only method that strikingly outperforms the other methods for this dataset is Crowd-Truth aggregation, which achieves a performance higher by almost ten points than all other methods in terms of F1 and CT F1. As this method is best considered a soft label method, we discuss this finding next.

With soft evaluation, silver training with D&S outperforms gold training both in learning the distribution of the annotations (i.e. evaluation using JSD and CE) and according to the Entropy Similarity metrics, but is outperformed by soft labelling methods.

### Learning from soft labels

The difference in F1 and CT F1 performance between CT ‘aggregation’ and all other crowd-only methods with this dataset is, we believe, due to the same reason which explains the better performance of D&S over MACE aggregation: the focus on the True class. D&S aggregation learns models of the coders’ sensitivity and specificity to the True class; the objective of CT aggregation is to find good examples for the True class.

However, other soft label methods apart from CT aggregation do not improve results over silver training when evaluating using the hard metrics. This is most likely due to the fact that this dataset does not satisfy any of the conditions under which soft label methods achieve good performance: it is the second smallest, and the quality of the annotations is the second lowest. However, soft label training does outperform hard label (silver and gold and augmented) training when evaluating using soft evaluation metrics.

### Filtering and weighting by inverse difficulty

One clear result for filtering and weighting on this task is that both approaches leads to significantly worse Accuracy than non-filtering/non-weighting using hard evaluation metrics. And, while filtering leads to better soft evaluation results, weighting largely remains on par with the non-weighting counterparts. Neither method leads to gold-level hard evaluation performance.

## 6.4.4 Recognizing Textual Entailment

The key characteristics of the RTE dataset are that it's the smallest dataset, counting only about 800 items, but it has a good number of annotations per item (10). The quality of the coders, as measured by average coder accuracy (0.84) and percentage of expert coders (0.83) is quite good, although not as high as that of IC-CIFAR10H in particular. The average number of annotations per coder is not very high however, at 48.78.

### Gold vs. Non-Gold

One obvious characteristics of RTE is that although using gold labels still yields the best hard evaluation results (with gold or gold+soft achieving the best results depending on the metric) the margin between training with gold labels and training with crowd labels only is much smaller than with the two datasets we have seen so far, POS and MRE–in fact, soft-loss and weighing methods achieve equivalent results to using gold with Accuracy and F1. (We cannot make a direct comparison to PDIS as that dataset has no gold labels.) This result was already reported by the creators of this dataset, Snow et al. [2008], but without explanation; we believe it can be straightforwardly explained in terms of quality of coders. In RTE, the coders have a much higher average accuracy with respect to gold labels, and the percentage of expert coders is very much higher, than with MRE in particular; as for POS, as discussed in Section 6.4.1, the headline coder accuracy and expert percentage figures are deceptive, in that accuracy is only high with one category, but for the other categories is pretty low. Further evidence for this explanation is the fact that the quality of aggregated labels is very high even though each annotator only produced relatively few annotations. The majority voting accuracy is already 90% with respect to gold (or 93% depending on how the ties are broken). The other hard aggregation methods also produce labels with 93% Accuracy. It is therefore not surprising that the margin between performance of gold training and crowd-based training is virtually nil.

### Using soft labels to supplement gold labels

The hard evaluation performance of gold-plus methods with RTE is slightly worse than that with MRE. Supplementing the gold labels with crowd information leads to non-significant improvements over gold training in terms of Accuracy, to equivalent results with the other metrics. This is due in our view to the fact that the diversity of the labels, as measured with BDE, is lower than with MRE (and half that with POS, where gold+soft methods do significantly improve over gold). As for soft evaluation, we find that MTLSL achieves significant improvements over gold in terms of JSD, but otherwise the results obtained supplementing gold labels with crowd information are comparable to those obtained training with gold labels alone.

### Learning from aggregated labels

Training using any silver label achieves slightly lower results with RTE than training with gold, less than one Accuracy/F1/ CT F1 points–a margin that is significant but much lower than that observed with POS and MRE. This would seem surprising given the relatively small number of annotations per coder, but as already discussed, we think it is due to the high quality of coders; this hypothesis is confirmed by the fact that MV achieves comparable results to the probabilistic aggregation methods. None of the silver label methods is significantly outperformed by any other method for learning from crowds only. For soft evaluation, in most cases, there is no significant difference between training using gold labels and training using silver labels: both gold and silver label training methods are outperformed by soft labelling and augmented methods.

### Learning from soft labels

When discussing the results with POS in Section 6.4.1 we pointed out how soft labelling training achieves as good or better results in terms of hard evaluation than aggregate labels with all datasets. Specifically, there are three datasets with which soft label training gives better results–POS, IC-LABELME, and IC-CIFAR10H–and three with which the results are equivalent–PDIS, MRE, and RTE. What characteristics do these last three datasets have in common?

In Section 6.4.1 we argued that training with aggregated labels matches performance with soft labels when average coder accuracy is relatively low, yet there are enough annotations per item and per coder to allow the aggregation method to acquire good models of the coders, resulting in high quality aggregated labels. We saw in Section 6.4.2 that these conditions hold for PDIS; they hold for RTE as well. They do not however hold for MRE.

But there are two additional characteristics in common to these three datasets. The first is that these three datasets for which soft labelling training doesn't improve over silver aggregate training are all binary classification tasks. It may be that in terms of hard evaluation, a model trained for binary tasks is always better off 'taking a stand' as opposed to taking a probabilistic approach to truth. Another characteristic these datasets have in common is that these are the datasets with the highest raw distribution entropy (see Table 2.1). Perhaps, soft-loss training is not tolerant to too much confusion. Digging a bit further we find that it is the soft loss methods that perform on par with silver training methods on this dataset; they outperform Repeated Labelling and DLC. In fact, the Repeated Labelling method proposed by Sheng *et al.* achieves on this dataset worse results than training using MV labels with all hard metrics–this is the only dataset for which this happens. RTE is also the dataset with the highest item entropy (0.72, 0.34 points higher than the next highest one, PDIS). Taken together, these facts suggest that the Sheng Repeated Labelling method is not suited for datasets with such characteristics. This hypothesis is further strengthened by the fact that the next method for which Repeated labelling achieves much worse results than the best silver or soft-loss method is PDIS, the dataset with the next highest entropy.

With soft evaluation, the results are somewhat mixed. However, we can definitively say that with all soft evaluation methods except for Jensen-Shannon Divergence, soft labelling methods always achieve the best results.

#### Filtering and weighting by inverse difficulty

As with the other datasets, filtering items with low agreement did not yield any improvements over training using all the items in terms of hard evaluation, and weighting by observed agreement did not achieve better results than using majority voting labels without weighing items. However, unlike what we observed with PDIS and with MRE, weighting using the inverse difficulty scores inferred by the Whitehill et al. [2009] aggregation model resulted in a substantially worse performance when evaluated using hard metrics.

### 6.4.5 Image Classification 1: LabelMe

IC-LABELME's most distinctive features are the low number of annotations per item (2.5 on average), the extremely low coder accuracy with respect to gold (.69 average accuracy, and only 42% of coders achieving expert accuracy levels), and the extremely high BDE (.76, almost double the next highest).

#### Gold vs. non-gold

With IC-LABELME we find again a large difference between methods using gold and methods using crowd information only. With hard evaluation, we find that using gold results in more than ten percentage points for Accuracy and F1 and slightly less for CT F1, similar to what we observed with MRE and POS. The same explanation we proposed for MRE and, after some analysis, for POS–that the reason for the large difference is the poor quality of annotators, or, more accurate, the substantial difference between their judgments and gold judgments—applies to IC-LABELME as well: these are the datasets with the lowest annotator accuracy and the lowest percentage of expert-quality annotators.

By contrast, with soft evaluation, the situation is exactly reversed, and we find a large difference with all soft evaluation metrics in favour of methods using crowd information, either by itself in soft labelling methods—in particular, soft loss methods, but also training with aggregated labels—or in combination with gold labels: MTLSL performs on par with soft-loss methods when measure by cross-entropy and JSD, and only slightly worse in terms of the entropy similarity measures.

This reversal confirms what was already obvious from the example from previous discussions, namely, that gold judgments are very different from crowd judgments in this dataset. The low accuracy of coders wrt to gold may then indicate the extreme subjectivity of judgments rather than carelessness–further evidence for this being provided by the extremely high BDE, by far the highest in any of the datasets we used.

### Using soft labels to supplement gold labels

As already mentioned regarding the discussion of gold vs. non-gold, very different results are achieved with this dataset by leveraging crowd information in addition to gold labels depending on which form of evaluation is used.

The best hard evaluation results for this task are obtained by training with gold labels alone: supplementing gold labels with crowd information leads to a decline in performance wrt to training with gold alone which is significant in all cases except with F1 evaluation of the MTLSL method; and the difference between gold only and augmented gold is only small with MTLSL.

But whereas crowd information doesn't improve upon gold for learning hard truth, MTLSL always significantly outperforms gold-only training with soft evaluation, and the other gold-plus training almost always do–the exception being the entropy correlation results, where Plank et al. [2014a] and MTLOA only remain on par with gold training. (Gold-plus methods are however generally outperformed by soft loss methods with this type of evaluation, except again for MTLSL that achieves equal-top performance with the soft-loss methods in terms of cross-entropy and JSD and near-top with the entropy correlation metrics.)

There are at least two clear reasons for this difference. First of all–although training over crowd information only can match or indeed outperform gold training, this only happens when certain conditions are met, as already discussed–which is not the case with IC-LABELME. In IC-LABELME, the average number of annotations per item is only 2.5, with a maximum of 3, and over 4% of the items only have a single annotation. In other words, the number of annotators per item is insufficient, so that the crowd annotations do not contain additional information for gold augmentation/regularization. And second, crowd judgments are very different from gold judgments with this dataset, as already noted above. As a result, methods relying on one type of judgments generally perform badly when evaluating against the other type, and viceversa–the one exception being MTLSL, which optimizes for both.

### Learning from aggregated labels

With this dataset, as with PDIS, probabilistic aggregation methods outperform majority vote aggregation by a substantial margin when evaluated against the gold label, and for the same reason: the poor quality of coders, or better the low similarity between their judgments and the gold, or between each other's judgments.

But while training with probabilistically aggregated labels outperforms MV, all silver methods are outperformed by soft labelling, weighting and filtering methods using all the evaluation metrics.

### Learning from soft labels

Soft-loss training significantly outperforms all other soft labelling methods with IC-LABELME in terms of hard evaluation, except for weighing and filtering (see next subsection). These soft loss methods also almost always produce the best soft evaluation results.

### Filtering and weighting by item agreement

IC-LABELME is the only dataset for which filtering items by observed agreement, and then training over the remaining items, leads to an improvement over training without pre-filtering. In fact, with this dataset filtering + MV labels is the best approach to learning from crowds. We believe that this is because this dataset is the one with the poorest quality annotators (or perhaps the annotators that disagree with most with the gold labels), as shown by the low OA figures and also by the fact this is the only dataset in which the expert annotators do not constitute a majority in the annotator population. Because the base model for this task was pre-trained with already learned and encoded images, the model loses nothing by discarding low observed agreement and perhaps mislabelled items.

## 6.4.6 Image Classification 2: CIFAR-10H

IC-CIFAR10H is not particularly big in size–it is comparable to POS or IC-LABELME–but it has very high annotator accuracy, with all annotators having an accuracy of 75% or more.The only other dataset with a percentage of expert coders this high is POS; but, unlike in POS, the annotators did not overwhelmingly label only one category. IC-CIFAR10H also has the highest number of annotations per item–over 50. Also, each coder annotated about 200 items on average. As a result of coder quality, high number of annotations per item, and good number of items per annotator, the quality of the aggregated labels is the highest, .99–and this irrespective of the type of aggregation used. Finally, this is a dataset with very high OA and very low entropy, both raw and BDE.

### Gold vs. non-gold

Another result of the high quality of coders and high number of annotations is that this is the one dataset with which training with crowd information outperforms training with gold labels, regardless of the method used. We already mentioned in connection with RTE the finding in [Sheng et al., 2008, Snow et al., 2008] that a large enough crowd may produce labels of quality comparable to that of gold labels produced by experts when the crowd workers are of sufficient quality; this dataset shows that in fact the crowd can outperform experts.

### Using soft labels to supplement gold labels

Two out the three methods for augmenting gold labels, the Plank et al. [2014a] method and MTLSL, result in significantly reduced hard evaluation performance with respect to gold on this dataset; only MTLOA achieves a performance on par with gold training. We saw the same result with IC-LABELME, and again we hypothesize that the reason is that the crowd annotations do not provide useful additional information for gold augmentation/regularization; but the reason is not the same as with IC-LABELME. With IC-LABELME, the motivation was the low number of annotations and the low quality of the annotators wrt gold. For IC-CIFAR10H, however, the reason is that the crowd annotations do not provide enough diversity in comparison with the gold labels, as they appear to be drawn from the same distribution; there is little disagreeent between gold labels and soft labels. This can be seen from the combination of high accuracy and high observed agreement of the crowd labels with respect to the gold. We can also see that both the raw annotation entropy and the BDE are extremely low, the lowest among all the datasets. Further evidence is that the gold+soft methods do not even outperform gold training in terms of soft evaluation–again, the only dataset for which this is the case.

### Learning with aggregated labels

D&S and MACE do not significantly improve over MV for this dataset, regardless of the evaluation metric. This is unsurprising given the quality of the coders–labels aggregated using majority voting already achieve the same accuracy (over 99% with respect to gold labels) as probabilistically aggregated labels accuracy, a sign that discriminating between annotators cannot offer much improvement over majority with respect to learning ground truth. For that matter, gold training and silver training are not significantly distinguishable.

### Learning with soft labels

As already said earlier, with IC-CIFAR10H we find that a large crowd providing high quality annotations can not only match, but outperform gold training. For this task, the soft labelling methods outperform all types of hard label training, both silver and gold. Among soft labelling methods, Repeated Labelling and DLC outperform soft loss and aggregated methods according to all the hard evaluation metrics, but soft-loss methods still outperform all hard-label methods, both gold and silver. The results with soft evaluation are more complex: Repeated Labelling and the soft-loss methods achieve the best results with cross-entropy by a wide margin over DLC, which however outperforms all other methods when evaluated using Jensen-Shannon Divergence. Soft-loss methods achieve the best results in terms of entropy estimation.

#### Filtering and weighting by item agreement

For this task, training with MV labels but weighting the loss for each item depending on the observed agreement for that item, leads to an improvement over majority voting training. This only happens with one other dataset, IC-LABELME, and the reason for this is not immediately apparent. In the case of IC-LABELME, one could argue that the dataset contains lots of hard items, as shown by the low overall agreement, and that the observed agreement works well at identifying confusing items. Removing items with below average observed agreement results in an average observed agreement greater than .98 for IC-CIFAR10H. The only other datasets for which this is true are PDIS, IC-LABELME and POS, and of the three, IC-LABELME is the only dataset/task in which the label of a given item is independent of the labels of the previous items. We conclude that MV with OA weighting is particularly suited for datasets with these characteristics.

Given the above discussion, one would expect that filtering would work for IC-CIFAR10H as well, but it does not. This is likely due to the fact that with filtering the model suffers from excessive data sparsity.

## 6.5 Discussion (Answering RQ5 and Revisiting RQ2)

### 6.5.1 Which method for learning from disagreement achieves the best results (RQ5)?

One of the key results of this PhD research is that the answer to **RQ5** is more complex that one would expect based on the previous literature. For one thing, new proposals do not unequivocally outperform previous proposals. A proposal like DLC while state-of-art is not the best method for training across *all* datasets with varying levels of disagreement. In fact, the results indicate that disagreement characteristics of the datasets (which reveal the nature of the task) and the form of evaluation (which is indicative of which model characteristics the ML practitioner/researcher values) are important consideration in choosing a method for learning from crowds.

Despite the complexity of the answer, one point stands out; soft labelling methods outperform hard labelling methods regardless of the level or source of disagreement and regardless whether hard or soft evaluation is used. Which soft labelling method performs best very much depends on the form of evaluation and the characteristics of the dataset. With soft evaluation, some form of soft-label training achieves best results with virtually all datasets and all metrics, except with Pearson correlation of entropy with rte, the smallest dataset. Specifically: some form of soft-loss training achieves the best results with all datasets except for rte and mre; Repeated Labelling achieves the best results with mre (all metrics) and rte (Entropy Similarity); and Deep Learning from Crowds generally achieves worse results than the other soft-label methods except with rte (ce) and ic-cifar10h (jsd).

With hard evaluation metrics, while soft loss training achieves competitive results for all the datasets, it is only the significantly best method for learning from crowds on pos, a dataset for which the average annotator (gold) accuracy is high but the accuracy of aggregated labels is unexpectedly low. For this dataset, as evidenced by the discussion in Subsection 6.4.1 and supported by Plank et al. [2014b], the disagreements are indicative of the complexity of the items for which annotators disagree, hence, soft loss training which informs the model of the possibility of alternative interpretation is well suited to learning on the datasets. For pdis, soft loss training using probabilistic soft labels from mace posterior has best results but this result is not significantly different that results from training with hard mace and d&s labels. Clearly pdis, which has a large number of coders of varying ability (and biases when labelling items subject to interface errors), benefits from the discriminatory power of probabilistic aggregation techniques.

For ic-labelme, pre-filtering low agreement items and training on the rest using Majority Voting Training, achieves the best results. On examining the filtered items, 60% of these were assigned labels *inside city*, *open country*, or *street* gold labels. These are categories for which the majority consensus disagrees the most with gold labels (see Figure 2.1), categories with the highest level of annotator confusion (see, Figure 6.4, and going by our analysis, these categories act as garbage categories as they are poorly defined and open-ended (see the discussion in Section 2.3.2). ic-labelme training benefits most from removing images for which the annotators detect overlap and hence disagree with the arbitrary gold interpretation. The next best results on hard evaluation of ic-labelme are obtained by mv + oa weighting which down-weights these items so that the model pays less attention to them during training.

mv + oa weighting also works well for rte; in fact, it has the best results amongst the non-gold methods, and as we noted, this is the dataset for which the annotator disagreements largely reflect the difficulty of the item. For ic-cifar10h, characterized by a high number of per-item annotations produced by high quality coders, Sheng Repeated Labelling and dlc methods, both of which multiply examples, outperform other methods. It is worth noting that although for this dataset annotators most resemble the gold, taking a soft approach to training is still beneficial over hard labelling; soft loss outperforms majority voting, dlc outperforms mace and d&s.

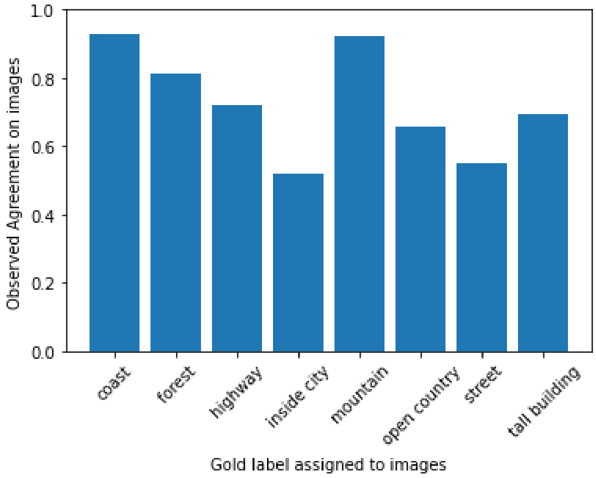


**Figure 6.4:** Observed agreement for images based on their gold categories

### 6.5.2 How can we best evaluate models on datasets that provide a range of judgments? (RQ2)

Finally, we return to **RQ2**. Throughout this discussion we have highlighted how much the relative ranking of the methods for learning with disagreement depends on the form of evaluation. The inevitable conclusion is that only using hard evaluation metrics such as Accuracy or F1, or only using soft metrics such as Cross-Entropy, will only provide a partial picture of how well a model performs on a dataset. So we would argue that only using a hard evaluation metric may only be considered appropriate for datasets on the low subjectivity end of the spectrum, the best predictor of which we found being what we called Best Distribution Entropy, or BDE. In all other cases, also reporting the results with a soft evaluation metric is arguably more accurate. At the other extreme, in the case of tasks where the labels are highly subjective, such as hate speech detection, it may be argued that using a hard metric makes little sense. On the other end, we can ask ourselves whether all of the metrics we studied in this thesis are required.

With regards to hard metrics, we can see that Accuracy, F1 and CT F1 rank the learning methods similarly, except on the MRE dataset which is highly imbalanced (see section 6.4.3). Also, while F1 and CT F1 rank the methods in a very similar way, we can see that the F1 metric increases the scores of all the methods. Comparing the F1 and CT F1 scores in Tables 6.2 and 6.3,[7] we can observe that for any given method, including the ones that do not take disagreement into account during training, the CT F1 score is always higher than the F1 score. This finding is consistent with the claim by Dumitrache et al. [2018c] that evaluating models under the assumption of a single correct answer underestimates a model's performance. We can also observe that using CT F1, the gains of highly accurate (typically, gold-trained) models over the less accurate models are reduced. This can be observed in the results with five of

[7] We compare CT F1 with F1 rather than with Accuracy as the two metrics differ only in the down-weighing of confusing items by CT F1 (see section 7.2.2)

the six datasets, the one exception being RTE. To appreciate this point, consider the difference between the best gold/gold-plus method and the best crowd-only method on the POS dataset in Tables 6.2 and 6.3. The F1 difference between gold training and training using the KL soft loss method is 11.14 points, while the CT F1 difference is 5.8. In other words, the difference we observed between the performance of the KL soft loss method and the performance of the gold model is much reduced when confusion is factored in. One interpretation would be that if we prefer to report only one metric, CT F1 may give a picture of the differences among methods less affected by hard items.

The differences among the results according to soft evaluation metrics are more substantial. Both CE and JSD measure the distance between the probability distribution outputted by a model and the target distribution, and the two measures are closely related, but apart from POS and IC-LABELME, these metrics yield very different results. More research is required in order to understand which of the two methods more accurately reflects intuition, but given that cross-entropy is already widely used in practice, it would certainly be reasonable to interpret our results as not providing sufficient reason for adopting JSD instead. The results with CE and JSD are also substantially different from those obtained with the two entropy-based measures, but these also differ with each other to a large degree. Again, no conclusion can be reached as to which of these metrics is more appropriate for the purposes of assessing how well models capture the uncertainty among human judgments.

## 6.6 Conclusion

In this Chapter we experimentally compared various methods (and approaches) to learning from the multiplicity of crowd annotations, by training key methods under each approach for all the tasks and evaluating these methods with soft and hard evaluation metrics discussed in Chapter 3. Our results suggest, first of all, that reaching a consensus on how to evaluate models if we abandon the gold standard is an essential prerequisite for this research, as the relative performance of the training methods under consideration is critically affected by the chosen evaluation. Our experiments do not allow us to reach a definitive conclusion in this matter as no metric was found to be more appropriate than any other. Until such consensus is reached, however, we found no issues with simply using cross-entropy to compare the output of a system to a soft label.

Secondly, regarding the identification of an overall ‘best’ approach, we observed a strong effect of dataset. With datasets of a substantial size and providing large numbers of judgments for each item, annotated by high quality coders, training directly from the soft labels achieved better results than training from aggregated labels, or even from gold labels, both when using hard evaluation and when using soft evaluation. When those conditions do not hold, leveraging gold labels generally achieved the best results in terms of hard evaluation. And leveraging soft labels in addition to gold labels generally achieved the best overall results, greatly improving performance when measured using soft metrics, and leading to as good or better results than using gold only in terms of hard evaluation with datasets not satisfying the conditions discussed above. Among the methods not relying on a gold label, it was notable that aggregation generally resulted in worse performance than training directly from the soft label, and particularly using repeated labelling or soft-loss methods.

## Chapter 7

# A Shared Task on Learning from Crowds

*Following the investigation into the different approaches to learning from crowds and how the performance of various methods are affected by the characteristics of the datasets, we organized a shared-task (competition)* `https://sites.google.com/view/semeval2021-task12/home`*. Participants were invited to submit novel models to learn these 6 datasets/tasks from crowds. This chapter summarizes the outcomes of the competition, relating them to the discussions and research questions.*

## 7.1 Motivation

The aim of the SemEval-2021 shared task on Learning with Disagreements (LE-WI-DI) was to provide a unified testing framework for methods for learning from data containing multiple and possibly contradictory annotations covering the best-known datasets containing information about disagreements for interpreting language and classifying images. The expectation being that unifying research on disagreement from different fields may lead to novel insights and impact AI widely. In this Chapter I describe the shared task and its outcomes.

## 7.2 Task Organization

In order to provide a thorough benchmark for methods for learning from disagreements, we used five well-known datasets for very different Natural Language ProcessingNLP and Computer Vision (CV) tasks, all characterized by providing a multiplicity of labels for each instance, by having a size sufficient to train state-of-the-art models, and by evincing different characteristics in terms of the crowd annotators and data collection procedure. We found or developed near–state-of-the-art models for the tasks represented by these datasets. Both 'hard' and 'soft' evaluation metrics - F1 and Cross Entropy (see Chapter 7.2.2).

The shared task was set up on the CodaLab Competitions platform,[1] which enables training and uniform evaluation on these datasets, such that the crowd learning adaptations of the base models proposed by participants to the task would be directly comparable.

In this section, we briefly introduce the five datasets. The dataset for Humour Preference Learning is described in detail. The datasets for the other four tasks (POS, PDIS, IC-LABELME, and IC-CIFAR10H) were introduced in Section 2.2, so they are merely outlined in this section, and the train:test:dev split used for this shared task is noted. For POS, IC-LABELME, and IC-CIFAR10H, the train:test:dev split differs from that used in the experiments in Chapters 4 to 6. This new split was necessary to provide 'true soft labels' soft evaluation in the development stage (since the development sets used in the preceding chapters did not contain crowd annotations). The new shuffle and split also serves to show that the results obtained in Chapters 4.2.3 to 6 are not peculiar to the particular partition used for those series of experiments.

This section also contains a description of the setup of the shared task.

### 7.2.1 Data

There are quite a few datasets preserving disagreements, and covering many levels of language interpretation; remarkably, none of these has ever been used for a shared task like this one, and the majority of them have never been used for a shared task at all. Our shared task has aimed at leveraging this diversity. The datasets included are outlined in this section. For a detailed outline of the first four datasets, see Section 2.2.

**The Gimpel et al. POS corpus** For GIMPEL-POS, consisting of over 14k examples (words/tokens) annotated by a median of 5 annotators per item, we selected 8.3K, 3K, and 3.1K tokens as training, development and test sets respectively. This is a departure from the split used in the previous experiments - 12K for training, 2.4k for testing and as development dataset introduced by Plank et al. [2014a] and used in the previous Chapters does not contain crowd labels.

**The PDIS corpus** The *Phrase Detectives* corpus [Poesio et al., 2019] is a crowdsourced coreference corpus collected with the *Phrase Detectives* gamified online platform [Poesio et al., 2013] [2]. We use PDIS, a simplified version of the corpus containing only binary information status labels. The training and development datasets consist of 473 documents (and 86.9K markables) and 24 documents (4.2K markables) respectively.

[1] https://www.microsoft.com/en-us/research/project/codalab/

[2] https://github.com/dali-ambiguity

#### The Humour dataset

The comprehension and appreciation of humour is known to vary across individuals [Ruch, 2008], making disagreement over the perceived funniness of jokes an appealing subject of study. For our training data, we used the corpus of Simpson et al. [2019], which consists of 4,030 short texts (3398 jokes, mostly based on puns, and 632 non-jokes such as proverbs and aphorisms). 28,210 unique pairings of these texts were presented to five annotators each, who indicated which text in the pair (if either) they found to be funnier. The goal is to learn a model that can predict binary pairwise labels that can predict which of two short texts is funnier.

The 4,030 text instances were split into 60% (2,418 texts, 9,916 unique pairs) for the training set and 20% (806 texts, 1,086 unique pairs) for the development set. Since this dataset has already been published, we constructed a new test dataset along similar lines: 1,000 short texts (all punning jokes) were paired in 7,000 different ways, and each of these 7,000 pairs was then presented to five crowd workers for a preference judgement[3].

#### The LabelMe dataset

Much research on learning from disagreements was motivated by computer vision datasets, so we intended to include some of these, too. We used the LabelMe dataset (see Section 2.2. For this shared task, we randomly selected 5K, 2.5K, and 2.5K images for training, development, and testing respectively, careful to keep the label proportions in each subset close to the proportions in the 10K dataset.

#### The CIFAR-10H dataset

We used CIFAR-10H dataset, the subset of CIFAR-10 for which Peterson et al. [2019] collected and median of 51 annotations per image (see Section 2.2). We randomly selected 7K, 1K, and 2K images for training, development and testing respectively. We kept as much data as we could for training without jeopardizing the evaluation process, as the base model was found to be sensitive to data size. As with the original dataset, each subset we created contains an equal number of images per category.

### 7.2.2 Evaluation metrics

As in the rest of this PhD research, the models submitted for this shared task were evaluated using both hard and soft evaluation metrics. Owing to the constraints of a shared task, we used the two most commonly used metrics across NLP and CV - Cross Entropy and F1 - the details of which are contained in Chapter 3. For the hard evaluation using F1, we evaluated the models' ability to predict the 'gold' label. And for soft evaluation using Cross Entropy, we evaluate the models by their ability

[3] US-based workers from Amazon Mechanical Turk were employed, paid in line with the federal minimum wage.

to predict a distribution similar to the *best soft label* - standard normalization soft labels IC-CIFAR10H and Humour Preference Learning, softmax soft labels for POS and IC-LABELME and the MACE posterior soft labels for PDIS (see Chapter 4 for discussions on the *best soft label*)

### 7.2.3 Task setup

CodaLab was the designated site for hosting SemEval-2021 competitions [4]. LE-WI-DI was run in two main phases:

**Practice phase.** In the practice phase, the goal was to train models for each task to learn from crowd annotations, given (1) the training data (consisting of raw and pre-processed input data and crowd annotations), (2) the development data with no labels, and (3) the base models (discussed in Section 7.3). While participants were encouraged to start with the base models and extend them, we did not make this mandatory. Participants could test the performance of their models on the development set by making predictions on the given development input data and then uploading their submissions to CodaLab for preliminary testing. We permitted up to 999 submissions in this phase. The 'leader board' was made public to allow participants not only to see how their models performed, but also to compare the performance of their model to those submitted by other participants.

**Evaluation phase.** The evaluation phase was the official testing phase of the competition. In this phase, we released test data (without labels) but we also released the gold labels and crowd annotations for the development set to facilitate quick offline testing and refining of models and model selection. The number of submissions for this phase was limited to ten submissions per participant to prevent the participants from fine-tuning their models on the test data.[5] The allowed number of submissions was later increased to 999 to more encourage submission attempts. The leader board was also kept public in this phase. Each participant could see the best model of each of the tasks using each of the evaluation metrics.

**Post-campaign evaluation.** As our aim was to make this benchmark available beyond the competition to researchers developing disagreement-aware models, we included a third, post-evaluation phase to allow lifetime access to the data. Researchers participating in this phase will be able to access the same data as in the evaluation phase and test their models on the test data for the various tasks.

[4] Our competition can be found at `https://competitions.codalab.org/competitions/25748`

[5] This proved unnecessary as the inherent difficulty of the shared task was enough of a deterrent.

## 7.3 Base Models

In order to encourage the participants to focus on the development of methods for learning from disagreement, as opposed to achieving higher performance by developing better models, we provided base models for each of the tasks represented by the aforementioned corpora. Details of the base models used for POS, IC-LABELME, IC-CIFAR10H can be found in Section 4.2.4. The paragraph below contains details for the model used for humour preference learning.

**The humour preference learning model.** We use as base model for this task Gaussian process preference learning (GPPL) with stochastic variational inference, as described and implemented by Simpson and Gurevych [2020]. As an input vector to GPPL, we first take the mean word embedding of a text, using 300-dimensional word2vec embeddings trained on the Google News corpus [Mikolov et al., 2013]. Then, we compute the frequency of each unigram in the text in a 2017 Wikipedia dump, and each bigram in the text in a Google Books Ngram dataset. Finally, we concatenate the mean unigram and bigram frequencies with the mean word embedding vector to obtain the input vector representation for each short text. The GPPL model is trained on pairwise labels from the training set to obtain a ranking function that can be used to score test instances or output pairwise label probabilities. As a Bayesian model, it takes into account sparsity and noise in the crowdsourced training labels, and moderates its confidence accordingly. Hence, it is a strong baseline for accounting for disagreement among annotators. This same GPPL approach set the previous state of the art on the humour dataset [Simpson et al., 2019].

## 7.4 Participating systems

Unfortunately, we observed a dramatic difference in the number of participants that signed up to the competition (over 100 groups), the number of groups that participated in the trial phase, and the number of groups that submitted a run for official evaluation.[6] Only one group, UOR, submitted in the evaluation phase [Osei-Brefo et al., 2021]. However, they did submit models for each of the tasks, and did adopt a learning from disagreements approach.

**POS tagging** For POS tagging, UOR developed a novel POS tagging model by fine-tuning the BERT language model Devlin et al. [2019]. The (tweet, token) pairs were encoded in the form

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

where the '[CLS]' token was added for classification and the '[SEP]' token separated the tweet from the token under consideration. To learn the class for the token, the

[6] Two participating groups cited an inability to come up with a novel crowd learning paradigm as the reason they did not submit for official evaluation.

learned classification token was passed through a single feed-forward neural network layer with softmax activation. The output of this layer represented the probabilities of the token belonging to each of the 12 classes.

To extend this model for crowd learning, UOR added an adaptation of the crowd layer from Rodrigues and Pereira [2018]. Rather than compute a single loss from the crowd layer as Rodrigues and Pereira [2018] do, UOR compute a joint loss from both the crowd layer and the base model (without the crowd layer bottleneck).

**PDIS classification.** For this task, UOR also used a fine-tuned BERT together with Rodrigues and Pereira's [2018] crowd layer. Each (document, markable) pair was encoded as follows:

[CLS] + Document + [SEP] + Markable + [SEP]

where the '[CLS]' and '[SEP]' tokens are used in the same manner as in POS tagging.

**Humour preference learning** For humour preference learning, the participant submitted predictions using the base model without modifications but made some changes to the training parameters [Osei-Brefo et al., 2021].

**LabelMe image classification (IC-LABELME).** For this task, UOR adapted the Rodrigues and Pereira [2018] crowd layer to the base model.

**CIFAR-10H image classification (IC-CIFAR10H).** For IC-CIFAR10H, the crowd labels were aggregated into hard labels using majority voting. However, UOR combined Zagoruyko and Komodakis's [2016] WideResNet model, which has been shown to outperform He et al.'s [2016] ResNet with the novel Sharpness-Aware Minimization (SAM) optimization technique, proposed by Foret et al. [2020], that has been shown to efficiently improve model generalization, especially on noisy, singly labelled data.

## 7.5 Results and discussion

### 7.5.1 A summary of UOR results

Table 7.1 contains the results of the participating system on this shared task when evaluated using the F1 metric with respect to the gold labels and the cross-entropy between the true soft labels for each task and the model prediction for that task). To place participant's result in context, we also report baseline results using two crowd learning approaches: majority voting and the soft loss (see Chapter 4). The best results for each task are highlighted in bold.

UOR concentrated their effort on the IC-CIFAR10H dataset, on which they did achieve good results and outperformed the baseline (see below). In the other datasets, their official results at the end of the evaluation phase were less competitive.

| Task | Model | F1 | Cross Entropy |
|---|---|---|---|
| POS | base model + MV | 0.753 | 2.263 |
| POS | base model + soft loss | **0.767** | **1.084** |
| POS | UOR (BERT + Crowd Layer) | 0.125 | 2.331 |
| PDIS | base model + MV | 0.906 | 0.397 |
| PDIS | base model + soft loss | **0.928** | **0.273** |
| PDIS | UOR (BERT + Crowd Layer) | 0.474 | 0.830 |
| HUMOUR | base model (GPPL) | **0.557** | **0.728** |
| HUMOUR | UOR | 0.513 | 3.697 |
| IC-LABELME | base model + MV | 0.806 | 2.833 |
| IC-LABELME | base model + soft loss | **0.833** | **1.691** |
| IC-LABELME | UOR (base model + Crowd Layer) | 0.784 | 1.769 |
| IC-CIFAR10H | base model + MV | 0.646 | 2.610 |
| IC-CIFAR10H | base model + soft loss | 0.698 | 1.052 |
| IC-CIFAR10H | UOR (WideResNet + SAM) | **0.769** | **0.827** |

**Table 7.1:** Results on the benchmarks and participant submissions on all the tasks using F1 (higher is better) and Cross Entropy (lower is better)

With the POS task, the model proposed by UOR, a BERT classification model with a modified crowd layer, achieved substantially worse results than training from a label aggregated using majority voting or training using a soft-loss function, both according to the hard evaluation metric (F1) and the soft metric (CE). While the ranking between soft-loss method, aggregation, and crowd layer with POS is consistent with that obtained in Chapter 6, the differential between soft-loss/MV training and the results obtained by UOR is much higher than anticipated given the result in Chapter 4.2.4. This suggests that the UOR's crowd layer is not effective as the DL-MW variant of Rodrigues and Pereira's [2018] crowd layer used in the previous Chapters. This suggestion is supported by UOR's result on IC-LABELME and PDIS - unlike the results on the DLC model in Chapters 4 and 6, UOR's crowd layer was outperformed by MV training for both tasks. The POS and PDIS results also suggest that the BERT model proposed by UOR yielded much lower results than the base models provided for those tasks.

For the humour preference learning task, again, the base model outperforms UOR's submission on both metrics,but in this case the difference in performance between GPPL and UOR is much less substantial with the hard metric, although it remains large according to the soft metric. As UOR's submission was also produced by the same base system, this large difference is possibly due to the choice of training hyper-parameters. A possible reason for poor cross-entropy error is the use of discrete labels, which are heavily penalized for overconfidence by cross-entropy error. On this soft metric, the Bayesian probabilistic approach of GPPL may have advantages over approaches with poorer calibration, which remains to be explored in future work. The GPPL approach therefore remains the state of the art with this dataset.

### 7.5.2 Learning IC-CIFAR10H from Noisy Single Labels

There is one dataset, however, on which UOR outperformed the two baselines: IC-CIFAR10H. For this dataset, the WideResNet image classifierUOR [Zagoruyko and Komodakis, 2016] and trained on majority voting aggregated labels and optimized using Foret et al.'s [2020] SAM optimization technique. The results show that WideResNet outperforms ResNet with this task both according to the hard metric and the soft metric. Interestingly, this is the one dataset in which the Deep Learning from Crowds approach of Rodrigues and Pereira [2018] works best as seen in Chapter 6, outperforming both soft-loss training and majority voting training. It would thus be interesting to understand if the performance of UOR's model could be further increased by adopting one of these methods.[7].

## 7.6 Conclusion

This shared task presented the first unified testing framework for learning with disagreements. The datasets include sequence labelling, three classification tasks, and preference learning, hence provide a test-bed for a wide range of challenges when learning from multiple annotators. We proposed to evaluate not just the 'hard' performance against a gold standard, but also the ability to predict the distribution of different interpretations of the data—that is, the alternative labelling provided by different annotators. The results show the benefit of soft loss functions that account for the distribution of labels in the training data. However, modelling alternative interpretations of data remains an under-researched topic in NLP and computer vision. To encourage future work on learning with disagreements, the shared task and datasets will remain available for evaluating new methods.

[7] As a postscript, we should note that after the end of the official competition we did carry out an investigation of the reasons for the poor performance of UOR's models on the tasks other than IC-CIFAR10H. Some points emerging from the discussion are presented in the participants' paper for the shared task

# Chapter 8

# What Models Know about Arbitrary Targets

*One of the observations emerging from this work is that different methods appear to work best depending on characteristics of the dataset such as the level of noise, for which different measures have been proposed. In this Chapter, I propose to learn the ground truth label and the model's opinion of the level of noise in an item via automated temperature scaling; learning a soft-loss function jointly with a temperature scaling parameter. We test this approach across five classification datasets with varying levels of noise from different sources. The results show that model calibration via automatic temperature scaling is a simple yet effective approach to learning accurate ground truth predictions in high disagreement datasets with overlapping labels and yields state-of-art results. Further, we analyze the model's per item temperature predictions and find that it correlates with several measures of noise such as reversed entropy and observed agreement.*

## 8.1 Introduction

As we discussed in Section 2.3 of Chapter 2, there are a number of reasons for this disagreement, ranging from *ambiguity*–items being interpretable in different plausible ways [Poesio and Artstein, 2005, Plank et al., 2014b, Poesio et al., 2019]–to *subjectivity*–different people having different views, e.g. on a particular text, e.g., on whether a review is positive or negative [Kenyon-Dean et al., 2018] or a tweet is offensive or not [Caselli et al., 2020]–to *overlapping labels* due to imprecise annotation schemes as in IC-LABELME – to *difficulty* experienced by annotators in annotating the item [Beigman and Beigman Klebanov, 2009], to simple *errors*, i.e., errors made by coders or caused by problems in the interface.

In this Chapter, we present preliminary results on a proposal, **automatic temperature-scaled soft loss**. This approach to learning with disagreement seeks to flatten or heighten the entropy of a model's predicted probability distribution depending on whether or not the model perceives the instance under consideration to have high **data uncertainty** resulting in arbitrary training targets or multi-modal label distributions. Section 8.2 outlines the methodology of this method. Section 8.3 provides the setup of the experiments conducting. Sections 8.4 and 8.5 discuss the results and analyze the temperature predictions of the model. Finally, Section 8.6 contains concluding thoughts.

## 8.2 Methodology: Temperature-scaled Soft Loss

In this Section, we explain the workings of our proposal, temperature-scaled soft loss - the combination of soft loss learning with automatic temperature scaling. Firstly, we recap the soft-loss function; then we extend the soft-loss proposal by including exploration of the suitability of various standard loss functions for soft-loss training. Finally, we detail the (automatic) temperature-scaled soft-loss methodology which involves weighting the soft loss for each item by a learned temperature parameter.

### 8.2.1 Soft Loss Learning; Finding the Suitable Loss Function

In Chapter 4, we defined a soft loss function as a standard (probability comparing) loss function targeting a soft label (generated from a crowd label distribution), $\mathbf{y}_i = p_{hum}$ rather than a hard gold or aggregated label. This concept was also explored by Peterson et al. [2019]. We also saw that the (gold) accuracy of the predictions made by a soft loss model is dependent on the method used in generating the probabilistic soft labels, which in turn is dependent on the annotation characteristics of the dataset. We examined two standard generation functions – the softmax function and the standard normalization function – and found that while standard normalization soft labels are preferable datasets like CIFAR-10H-10H annotated by by a large number of gold-quality with high observed agreement a high-agreement datasets like dataset annotated annotators, the softmax loss function is more suitable for datasets that do not meet these criteria like Gimpel et al.'s POS and LabelMe. [Uma et al., 2021b] further showed that for mixed quality datasets like PDIS, the best generation method is using a probabilistic aggregation model like MACE [Hovy et al., 2013]. We then designated these best performing soft labels to be our 'best soft labels' which we use in the rest of the experiments.

#### A Suitable Loss Function

Although we state that the loss function used in soft loss training can be any probability comparing loss function, we constrained to use the cross entropy loss function following the hypothesis by Peterson et al. [2019], that it was uniquely suitable for the task. Recently, Malinin and Gales [2018] showed that for datasets with high data uncertainty resulting from class overlap and leading to a multi-modal label distribution, the reverse KL-divergence function is the appropriate loss function if the goal is to maximize prediction accuracy. They test their hypothesis on synthetic data, comparing the reverse KL-divergence loss function with the (forward) KL divergence function and show that while a KL-divergence loss function is a sensible choice for datasets with low data uncertainty, the reverse KL-divergence is more suitable when this is not the case.

Thus, a preliminary experiment, we test the hypothesis of Malinin and Gales by training soft-loss functions for each task using the *best soft label* and each of the divergence functions. We additionally test the other two well known probability-comparing loss functions - the cross entropy loss function (CE) already used in the previous Chapters and the Mean-squared error function (MSE). Soft loss functions using each of the stated functions can be expressed using simplified notation:

- Cross Entropy Soft loss[1]:

$$CE(y_{hum},\ y_\theta) = -\sum_{i=1}^{n} y_{hum}^i \log y_\theta^i \tag{8.1}$$

- KL Soft loss:

$$D_{KL}(y_{hum} \mid\mid y_\theta) = \sum_{i=1}^{n} y_{hum} \log(\frac{y_\theta^i}{y_{hum}^i}) \tag{8.2}$$

- Reverse KL Soft loss:

$$D_{RKL}(y_\theta \mid\mid y_{hum}) = D_{KL}(y_{hum} \mid\mid y_\theta) = \sum_{i=1}^{n} y_\theta^i \log(\frac{y_{hum}^i}{y_\theta^i}) \tag{8.3}$$

- MSE Soft loss:

$$MSE(y_{hum}, y_\theta) = \sum_{i=1}^{n} (y_{hum}^i - y_\theta^i)^2 \tag{8.4}$$

where $y_{hum}^i$ is the target label for an item $i$, the *best soft label*; $y_\theta^i$ is the model's predicted probability distribution for that item; and $n$ is the number of items in the training set.

We experiment with these variations of the soft loss function and note the prediction accuracy of the trained models, especially in reaction to Malinin and Gales's [2018] hypothesis. The best soft loss function is used for experiments in automatic temperature scaling.

### 8.2.2 Item Weighting through Automatic Temperature Scaling

*Automatic temperature scaling* combines ideas from both *temperature scaling* and *Platt scaling.* Platt scaling was proposed to calibrate a logistic regression model i.e. adjust its parameters to reflect uncertainty [Platt, 1999]. To calibrate a model, Platt proposes that two scalar parameters, a and b ∈ R, be learned by optimizing the negative log-likelihood function over the validation set while keeping the model's parameters fixed.

[1] observe that the reverse KL reverses the direction of the forward KL divergence function

The learned parameters are used to rescale the logits of the model, $\mathbf{z}_i$ resulting in outputs, $f(\mathbf{x}_i) = \sigma(a\mathbf{z}_i + b)$.

Temperature scaling is single parameter variant of Platt scaling [Guo et al., 2017], where the same scalar parameter, $T$, called the temperature, is used to rescale logit scores for all the classes, $\mathbf{z}_i$, before applying the softmax function. This way, the model's recalibrated probabilities are given as:

$$f(\mathbf{x}_i) = \sigma(\mathbf{z}_i/T) \tag{8.5}$$

where $\sigma(\cdot)$ is the softmax function. $T > 1$ raises the entropy of the output probabilities, hence "softening the softmax" and evening out the probability distribution; $T < 1$ hardens the softmax resulting in a peakier (more modal) probability distribution; and $T = 1$ recovers the unscaled probabilities [Guo et al., 2017]. The value of $T$ is obtained by minimizing the negative log likelihood on a held-out validation dataset. Because $T$ is independent of the class, $j$ and the item, $i$, *temperature scaling does not affect which class is predicted and hence does not affect prediction accuracy.*

*Automatic temperature scaling* is a natural extension of temperature scaling. It differs from temperature scaling in three key ways. Firstly, rather than optimizing $T$ on a held-out validation set, automatic temperature scaling learns parameter $T$, jointly as it learns to predict the classes. It does this by learning a network of weights $\mathbf{w}_T$ and biases $b_T$ such that

$$T_i = softplus(\mathbf{W}_T\mathbf{x}_i + b_T) \tag{8.6}$$

This expression of temperature is similar to *matrix scaling*, an alternative temperature scaling proposal of Guo et al. [2017] [2] and also similar Platt scaling which also learns two parameters instead of one but unlike both methods, the parameters are not tuned on a held-out validation set. Rather, during training, the model's outputs, $\hat{y_i} = f(x_i)$ are computed as:

$$f(\mathbf{x}_i) = \sigma(\mathbf{z}_i * T_i) \tag{8.7}$$

and the model's loss is computed using the appropriate soft loss function. In this way, the model jointly learns classifier and scaling parameters.

The second key difference is practical in nature but has notable implications. Unlike temperature scaling where the logits are divided by the temperature, $T$, in automatic temperature scaling, the logits are *multiplied* by the temperature as we found this to work better in practice. The implication is that in automatic temperature scaling (and conversely to temperature scaling), a warmer temperature (higher values of $T$) indicate lower uncertainty resulting in peakier probabilities while colder temperatures indicate higher uncertainty resulting in a more even distribution.

The third key difference can be observed from the definition of $T_i$ in Equation 8.5. Unlike temperature scaling, the model does not have a single temperature value,

[2] Guo et al. propose the use of the $max(\cdot)$ function, rather than $softplus(\cdot)$

rather, the temperature of any given item is a function of the input vector for the item and the temperature weights of the model, $W_T$ - the logits for each instance are scaled a different temperature, determined by the model and learned as a function of the input features of the instance. We hypothesize that if the model is able to perceive uncertainty from the input data, it will respond by producing a lower temperature value for that item. The converse is also true. Thus, by considering each instance separately, the model is able to produce temperature values depending on how much data uncertainty it perceives for each item.

This third key is vital to understanding the anticipated improvement in predictive accuracy using automatic temperature scaling. Lowering the temperature on instances with high data uncertainty (i.e. instances perceived to be prone to overlapping labels) will result in a flatter distribution for such items, increasing the loss contribution of that item to the overall loss. This both draws the model's attention to such items and adds arbitrariness of the target labels for such items, a desired effect as overlapping labels imply arbitrary targets. For a low data uncertainty instance, the model would produce a higher temperature resulting in a peakier distribution, reflecting how confident the model is in the absoluteness of the target label.

## 8.3 Experiment Setup

We conduct the experiments in this Chapter in two phases. Firstly, we experimentally compare the suitability of various standard loss functions for soft loss training as outlined in Section 8.2.1 on several tasks. Then, we extend the best performing loss function into an automatic temperature-scaled soft loss. For both experiments, we evaluate the models based solely on predictive accuracy.

Following the observations from Chapter 6, we restrict our experiments to the tasks with the larger datasets - POS, PDIS, IC-LABELME, and IC-CIFAR10H. The training details for the base models are the same as we outline in Chapter 4.

## 8.4 Results

Table 8.1 contains results accuracy results of comparing the performance of different probability-comparing loss functions for making gold predictions. And, Table 8.2 outlines the results on adding automatic temperature scaling to the best soft loss function from Table 8.1, also evaluated using prediction accuracy. As in the other experiments, we measure significance via bootstrap sampling, following Berg-Kirkpatrick et al. [2012] and Søgaard et al. [2014]. The rest of this section discusses the results from these tables, noting the significant results.

### 8.4.1 Choosing the loss function

In Section 2.3 of Chapter 4, we highlighted the Russell et al.'s [2008]'s LabelMe dataset as the dataset for which the primary cause of disagreement is overlapping labels. If

|  | POS | PDIS | IC-LABELME | IC-CIFAR10H |
|---|---|---|---|---|
| MSE soft loss | 79.20 | 92.90 | 84.21 | 63.49 |
| CE soft loss | 79.80 | 92.86 | 84.66 | 66.54 |
| KL soft loss | **79.96** | 92.86 | 84.73 | **66.58** |
| Reverse KL soft loss | 79.81 | **92.95** | **84.97** | 63.71 |

**Table 8.1:** Different Loss Functions for Soft Loss Training and their effect on Accuracy

Malinin and Gales's hypothesis that reverse KL divergence is uniquely suitable loss function for training on such datasets, we expect to see that on this dataset, reverse KL soft loss would have the highest accuracy of all the soft loss function; and indeed we do. From Table 8.1, we see that training with reverse KL as a loss function outperforms all other soft loss function by at least 0.25 accuracy points on IC-LABELME, though this margin is not significant.

For PDIS, reverse KL also remains on par with KL divergence according to significance tests, only outperforming it by a 0.9 point margin. For POS and IC-CIFAR10H, reverse KL soft loss do not outperform forward KL soft loss in-fact for IC-CIFAR10H falls nearly 3 significant points below KL soft loss. These results are not surprising given the disagreement analysis carried out in Section 2.3 and the dataset characteristics shown in Table 2.2.6; our disagreement analysis do not reveal the predominance of overlapping labels in the POS and IC-CIFAR10H datasets, and the table shows that in these two datasets of the four used in this Chapter, annotators have near-gold accuracy.

The results also show that while CE soft loss remains on par with forward and reverse KL soft loss, MSE falls below the other functions in every task but the binary classification task MSE. This again, is unsurprising as MSE pays a lot of attention to the other classes besides the modal class, and for a task like CIFAR-10H shown to have the least amount to disagreement on what the modal class should be, this is undesirable. Hence, following from these results, we use KL soft loss as the starting point for automatic temperature-scaled soft loss for POS and IC-CIFAR10H; and reverse KL soft loss for PDIS and IC-LABELME.

### 8.4.2 Temperature Scaling Soft-Loss Learning

The first point we make in Table 8.4 is that automatic temperature scaling significantly improves upon soft loss training in one task: IC-LABELME. This confirms our hypothesis that when the main source of disagreement is fully observable from the input, the model is able to learn useful information and calibrate itself through automatic temperature scaling. In other words, automatic temperature scaling works when the source of disagreement is data uncertainty brought on by overlapping labels and resulting in the arbitrariness of ground truth.

On POS and PDIS, the effect of temperature scaling on the POS and PDIS models is not significant. These are the datasets for which we as well as Plank et al. [2014b] and Poesio et al. [2019] have shown that the disagreements in these datasets are

| Task | Model | Accuracy |
|---|---|---|
| POS | KL soft loss | 79.96 |
| POS | KL soft loss + $T_i$ | **80.01** |
| PDIS | Reverse KL soft loss | 92.95 |
| PDIS | Reverse KL soft loss + $T_i$ | **93.00** |
| IC-LABELME | Reverse KL soft loss | 84.97 |
| IC-LABELME | Reverse KL soft loss + $T_i$ | **86.51** |
| IC-CIFAR10H | KL soft loss | **66.58** |
| IC-CIFAR10H | KL soft loss + $T_i$ | 63.89 |

**Table 8.2:** Results showing the Accuracy and F1

largely due to linguistic ambiguity and/or interface limitations. As stated in these works, linguistic ambiguity is often contextual and implicit and hence neither linearly separable from other sources of disagreement nor fully observable Plank et al. [2014b], Poesio et al. [2019]. Therefore, it is not surprising that the models do not benefit from automatic temperature scaling. For IC-CIFAR10H with 0.92 observed agreement, we showed that the disagreements are due to difficulty experienced by annotators when labelling blurry images. These disagreements are not systematic or a result of an imprecise annotation scheme.

It is worth noting that recalling the results from Table 6.1 in Chapter 6, temperature-scaled soft loss outperforms state-of-art systems like Rodrigues and Pereira's DLC on IC-LABELME, and is at least on-par with the best disagreement aware method from the table i.e. Filtering.

## 8.5 Interpreting $T_i$

In this section, we examine the temperature predictions of the model to understand what the model knows about label arbitrariness.

One way to do this is to measure the correlation of the temperature values to known measures of item agreement/uncertainty/difficulty. Figure 8.5 shows the [Pearson, 1896] correlation of temperature to two of such metrics used throughout this research - observed agreement and normalized entropy.

The results show that for IC-LABELME, the only dataset for which our method produces a significant improvement over the soft-loss baseline, the model's $\xi$ predictions has the strongest positive correlation to observed agreement. This means that the model tended to make higher $\xi$ predictions for items with high observed agreement and lower $\xi$ predictions for items with a low observed agreement.

Since we posit that automatic temperature scaling works on LabelMe dataset because it contains overlapping labels, we also examine the label distribution of the instances with the lowest temperature predictions, i.e. the label distribution of the instances the model considers most arbitrary. Figure 8.3 is a bar chart showing this distribution. We compare this figure with the confusion matrix between the majority and the gold (shown in Chapter 4 and repeated here in Figure 8.2 for easy access).

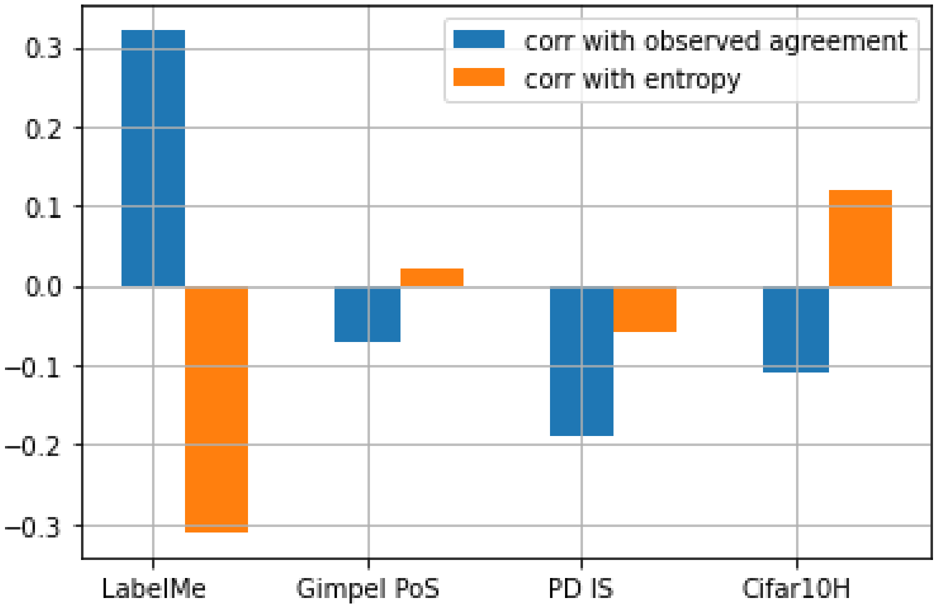
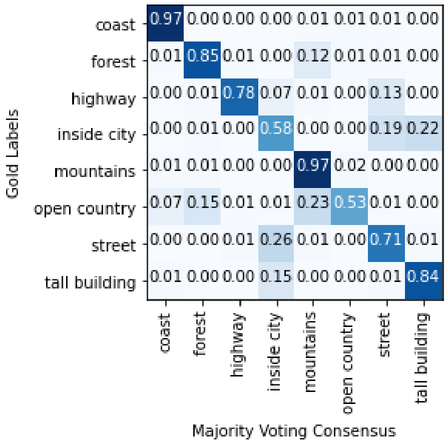


**Figure 8.1:** Graph showing the correlation of $T_i$ with observed agreement, entropy and percentage agreement with gold.



| | coast | forest | highway | inside city | mountains | open country | street | tall building |
|---|---|---|---|---|---|---|---|---|
| coast | 0.97 | 0.00 | 0.00 | 0.00 | 0.01 | 0.01 | 0.01 | 0.00 |
| forest | 0.01 | 0.85 | 0.01 | 0.00 | 0.12 | 0.01 | 0.01 | 0.00 |
| highway | 0.00 | 0.01 | 0.78 | 0.07 | 0.01 | 0.00 | 0.13 | 0.00 |
| inside city | 0.00 | 0.01 | 0.00 | 0.58 | 0.00 | 0.00 | 0.19 | 0.22 |
| mountains | 0.01 | 0.01 | 0.00 | 0.00 | 0.97 | 0.02 | 0.00 | 0.00 |
| open country | 0.07 | 0.15 | 0.01 | 0.01 | 0.23 | 0.53 | 0.01 | 0.00 |
| street | 0.00 | 0.00 | 0.01 | 0.26 | 0.01 | 0.00 | 0.71 | 0.01 |
| tall building | 0.01 | 0.00 | 0.00 | 0.15 | 0.00 | 0.00 | 0.01 | 0.84 |

**Figure 8.2:** Confusion matrix between gold labels and majority voting consensus for LabelMe

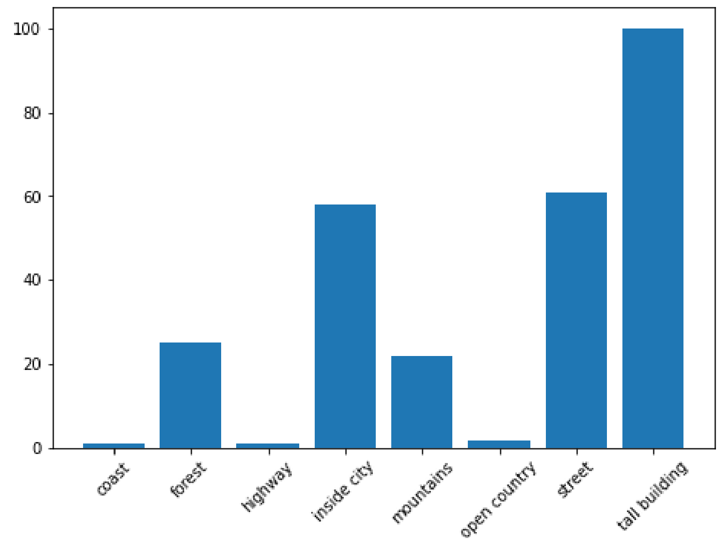


**Figure 8.3:** Bar chart showing the gold label distribution of the images with the lowest temperature i.e. images in the 1st quartile range of temperature

Considering both figures, we can see that the model captures some of the confusion of the annotators; the model assigned the lowest temperatures to images belonging mostly to the *tall building*, *street*, or *inside city*, capturing the overlap between these three categories.

## 8.6 Conclusions & Future Work

In this this we reported on the use of temperature scaling in a learning-from-disagreements setting. Our results show that model calibration via automatic temperature scaling is a simple yet effective approach to learning accurate ground truth predictions in high disagreement datasets with overlapping labels and yields state-of-art results. Further, we analyze the temperature values of the successful model to find that the temperature values have some correlation with two known measures of item disagreement/uncertainty – a positive correlation of about 0.3 with observed agreement and a negative correlation of about 0.3 with entropy. We also observe that the model assigns the lowest temperature to instances with one of the three labels *inside city, street, tall building* which we show in Section 2.3 to be overlapping. Future work would involve further probing the model to quantify what it captures in its temperature predictions. We will also investigate the effect of temperature scaling on more 'subjective' tasks like hate speech detection.

# Chapter 9

# Conclusion

## 9.1 Summary of our Contributions

With the growth in size and quality of annotated resources, and the increasing practice of employing several annotators to produce them, the idealization that a 'gold' interpretation can be specified for every item in the dataset and used as target during learning / evaluation, underlying much practice in supervised learning, has become untenable. Also increasing is the evidence that training with noisy labels results in better performance of the obtained models on unseen data; and the realization that for several observer dependent/subjective tasks, the gold standard is an arbitrary target. Abandoning this idealization requires the development of new paradigms both for training and for evaluating models. This research makes several contributions that are useful in this line of research.

Firstly, in Chapter 2, we examine crowd collected datasets for several AI tasks and show (with evidence and with reference to an extensive literature) that to varying degrees the gold standard idealization does not hold for these datasets. Further, we also examine various proposals for training models for these tasks and create a taxonomy of these method based on their handling of disagreement.

Having shown the evidence for disagreement as more than noise, we present a case for soft evaluation, an alternative to hard evaluation– evaluating models with an assumption a gold standard. We highlight several hard and soft evaluation metrics proposed in literature – accuracy, F1, CT F1, cross entropy, Jensen-Shannon Divergence and propose two of our own – entropy similarity and entropy correlation; and in doing this we compare the metrics and make a case for soft evaluation.

In Chapter 4, we address the question of whether models trained without the use of gold standard labels can compete with models trained on (the multiplicity of opinions available in) crowd data. We find that although the answer depends very much on the form of evaluation used, under certain conditions, models trained without assuming a gold truth can achieve better performance than models that leverage gold labels across all forms of evaluation. Our results and analysis revealed that when a dataset is annotated by a large number of high quality coders who agree amongst themselves, training a disagreement-aware soft loss model from the variety of crowd labels produces better results than training on gold labels. However, even when these conditions are not met, we find that training models with an awareness of disagreement is overwhelmingly better than training using a crowd consensus (an aggregate/silver label).

In Chapter 5, we answer the question 'Can information from crowd annotations be used in conjunction with gold labels to build better models compared to learning from gold labels only?' by proposing two multi-task methods that do this - MTLSL and MTLOA. Each of these proposals learn the gold label as a main task but additional information is provided to the model by means of an auxiliary task; learning the probability distribution of over the crowd annotations (for MTLSL) or learning the instance confusion as defined by observed agreement (for MTLOA), thus incorporating useful information about label uncertainly even while learning the prescribed/preferred interpretation. We found that while MTLOA remains largely equivalent to training on gold alone for hard evaluation, it often outperforms gold training when evaluated using soft metrics. The MTLSL however was shown to outperform gold training on a majority of tasks, using hard and soft evaluation metrics, thus providing evidence that even when the goal is to learn the gold labels, crowd information is still very useful.

In Chapter 6, we carried out an extensive experimental comparison of the approaches to learning from crowds, asking the question 'what is the absolute best method for leveraging crowd information?'. In answer to this question, we find a strong effect of dataset (disagreement) characteristic; there isn't an overall best method across tasks and datasets, rather the degree to which each approach works on a given dataset depends largely on the nature of the tasks reflected in the disagreement characteristics of the task. Overall, we found that soft labelling approaches overwhelmingly outperform hard labelling approaches across datasets using a variety of evaluation metrics. Regarding the choice of ideal evaluation metric, our experiments do not allow us to reach a definitive conclusion in this matter as no metric was found to be more appropriate than any other; perhaps the choice of metric is dependent on the target use of the trained models. Until such consensus is reached, however, we found no issues with simply using cross-entropy to compare the output of a system to a soft label.

## 9.2 Recommendations

In terms of outcomes from this thesis, the results suggest that the best way to achieve high-quality and empirically grounded datasets is to collect a substantial number of judgments from high-quality coders. This is what we observe from learning from the IC-CIFAR10H dataset—see Section 2.2.5 for a discussion of this dataset and Section 6.4.6 for a discussion of experiments on IC-CIFAR10H. With a dataset meeting such standards, the soft loss function would be the recommended approach, providing the most computationally efficient model with competitive hard and soft results: The Sheng et al. soft labelling approach while producing a slightly higher accuracy computationally more expensive due to repeated labelling; and DLC produces one of the worst cross entropy scores due to it's exaggerated trust (and higher weighting) of judgements by annotators it deems most reliable annotators (the majority of the coders).

For researchers working with already curated datasets that do not meet this standard, the choice of model and approach would be dependent on the evaluation goals or the intended end-use for the model. If the goal is to train a model to predict gold labels, such a model would be evaluated using a hard metric like accuracy or F1 depending on the need to control class-imbalance and the best form of training would be depend on the availability of gold labels for training. When gold labels are available and the crowd annotations represents a rich but not stochastic diversity of opinions (as indicated by a moderate best distribution entropy (BDE) as observed in POS, MRE, RTE), training on both gold and crowd labels using MTLSL offers the best of both worlds – competitive hard and soft results. However, if the task is poorly defined in a way that gives rise to random labelling (as indicated by a combination of a high BDE coupled with low alignment between the gold and the crowd as measured by average accuracy of annotators with respect to gold) as in IC-LABELME in Section 6.4.5, training on gold labels alone would provide the best results. Where the dataset does not meet the IC-CIFAR10H standard and gold labels are unavailable, the soft loss approach offers the best results using both hard and soft evaluation metrics. If the crowd is of mixed quality, the soft labels should be derived using a probabilistic aggregation method; otherwise, softmax soft labels offer the best hard evaluation results.

These recommendations are summarized in the somewhat simplified decision tree in Figure 9.1.

## 9.3 Future Directions

More and more the questions we put forth in this research are being asked in the larger research community. To consolidate efforts in this regard, we proposed and carried out a shared task on 'learning with disagreement', the proceedings of which we include as Chapter 7 of this thesis. By consolidating several datasets into one framework and allowing for hard and soft evaluation across these datasets, we anticipate that further work will be carried out to further address our research questions.

We also carry out preliminary experiments in another direction. In Chapter 8, we propose the automatic temperature scaled soft loss function, a model that can itself predict the arbitrariness of the target given the input and the probabilistic soft label. We find that our model's intuition about how confusing an item is not highly correlated to known metrics of annotator confusion (observed agreement or entropy). Rather, our model's confusion scores (temperature) is indicative of label/category overlap. This model shows promise in the direction of item uncertainty detection.

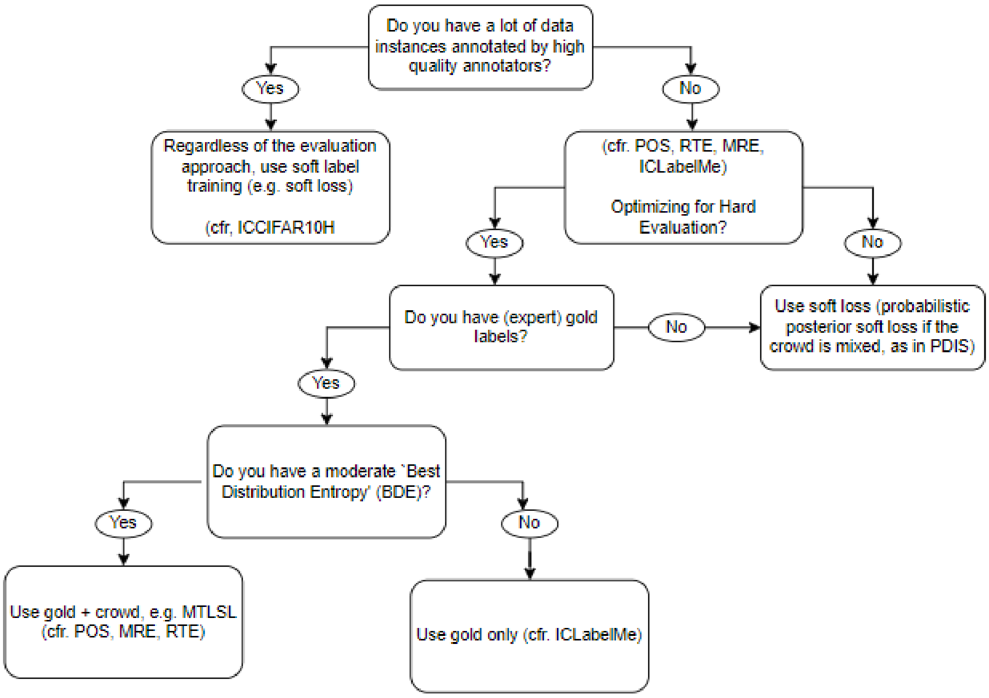

Do you have a lot of data instances annotated by high quality annotators?
Yes
No
Regardless of the evaluation approach, use soft label training (e.g. soft loss)
(cfr, ICCIFAR10H
(cfr. POS, RTE, MRE, ICLabelMe)
Optimizing for Hard Evaluation?
Yes
No
Do you have (expert) gold labels?
No
Use soft loss (probabilistic posterior soft loss if the crowd is mixed, as in PDIS)
Yes
Do you have a moderate `Best Distribution Entropy' (BDE)?
Yes
No
Use gold + crowd, e.g. MTLSL (cfr. POS, MRE, RTE)
Use gold only (cfr. ICLabelMe)

**Figure 9.1:** A guide to choosing a best-performing model given the characteristics of one's dataset and hard or soft evaluation

# Glossary

**best-performing distribution** the probability distribution generated from crowd labels using the extraction approach that results in the most accurate soft-loss model. 27

**crowdsourcing** the collection of annotations from a group of people, usually via the internet. 12

**disagree** to differ in judgement. 13

**disagreement-aware crowdsourcing** an annotation technique where annotators are asked to choose all labels that apply rather than a single label for each item. 23

**expert** an individual proficient at a (n annotation) task and motivated either by altruism or a financial incentive to provide labels for data instances that exemplify the task. Expertise in annotation tasks is often an acceptability judgement made by of the data. 12

**game-with-a-purpose** an annotation task redressed as a game or imbued with elements of a game so as to reduce the drudgery of the task. 12

**gold assumption** the assumption that a single preferred interpretation (an objective truth) exists for each instance to be annotated; and that where available, the gold label captures this objective truth. 13

**gold label** the final consensus of expert judgements usually obtained either by thoroughly discussing disagreements until a resolution is reached or by aggregating expert judgements. 12

**implicit ambiguity** ambiguity emerging from disagreements among annotators, rather than from annotators explicitly marking items as ambiguous. 20

**microtask crowdsourcing** recruiting a crowd of people to annotate small parts of a larger overarching task by offering small financial pay-outs. 12

# Bibliography

Sohail Akhtar, Valerio Basile, and Viviana Patti. A new measure of polarization in the annotation of hate speech. In *AI*IA - XVIIIth International Conference of the Italian Association for Artificial Intelligence*, Lecture Notes in Computer Science, page 588–603. Springer, 2019. doi: https://doi.org/10.1007/978-3-030-35166-3.

Shadi Albarqouni, Christoph Baur, Felix Achilles, Vasileios Belagiannis, Stefanie Demirci, and Nassir Navab. Aggnet: Deep learning from crowds for mitosis detection in breast cancer histology images. *IEEE Transactions on Medical Imaging*, 35 (5):1313–1321, 2016. doi: 10.1109/TMI.2016.2528120.

Lora Aroyo and Chris Welty. Truth is a lie: Crowd truth and the seven myths of human annotation. *AI Magazine*, 36(1):15–24, 2015. doi: 10.1609/aimag.v36i1.2564. URL `https://ojs.aaai.org/index.php/aimagazine/article/view/2564`.

Ron Artstein and Massimo Poesio. Survey article: Inter-coder agreement for computational linguistics. *Computational Linguistics*, 34(4):555–596, 2008. doi: 10.1162/coli.07-034-R2. URL `https://aclanthology.org/J08-4004`.

Valerio Basile. It's the end of the gold standard as we know it. on the impact of pre-aggregation on the evaluation of highly subjective tasks. In *Proc. of the AIXIA Workshop*. Universitá di Torino, 2020.

Valerio Basile, Michael Fell, Tommaso Fornaciari, Dirk Hovy, Silviu Paun, Barbara Plank, Massimo Poesio, and Alexandra Uma. We need to consider disagreement in evaluation. In *Proceedings of the 1st Workshop on Benchmarking: Past, Present and Future*, pages 15–21, Online, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.bppf-1.3. URL `https://aclanthology.org/2021.bppf-1.3`.

Eyal Beigman and Beata Beigman Klebanov. Learning with annotation noise. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 280–287, Suntec, Singapore, 2009. Association for Computational Linguistics. URL `https://aclanthology.org/P09-1032`.

Beata Beigman-Klebanov and Eyal Beigman. From annotator agreement to noise models. *Computational Linguistics*, 35:495–503, 2009. doi: 10.1162/coli.2009.35.4.35402.

Beata Beigman Klebanov and Eyal Beigman. 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, 2014. Association for Computational Linguistics. doi: 10.3115/v1/P14-2064. URL `https://aclanthology.org/P14-2064`.

Beata Beigman Klebanov, Eyal Beigman, and Daniel Diermeier. Analyzing disagreements. In *Coling 2008: Proceedings of the workshop on Human Judgements in Computational Linguistics*, pages 2–7, Manchester, UK, 2008. Coling 2008 Organizing Committee. URL `https://aclanthology.org/W08-1202`.

Taylor Berg-Kirkpatrick, David Burkett, and Dan Klein. 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, 2012. Association for Computational Linguistics. URL `https://aclanthology.org/D12-1091`.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal, 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1075. URL `https://aclanthology.org/D15-1075`.

Dallas Card, Peter Henderson, Urvashi Khandelwal, Robin Jia, Kyle Mahowald, and Dan Jurafsky. With little power comes great responsibility. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9263–9274, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.745. URL `https://aclanthology.org/2020.emnlp-main.745`.

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

Rich Caruana. Multitask Learning. *Machine Learning*, 1997. ISSN 08856125.

Tommaso Caselli, Valerio Basile, Jelena Mitrović, Inga Kartoziya, and Michael Granitzer. I feel offended, don't be abusive! implicit/explicit messages in offensive and abusive language. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 6193–6202, Marseille, France, 2020. European Language Resources Association. ISBN 979-10-95546-34-4. URL `https://aclanthology.org/2020.lrec-1.760`.

Robin Cooper, Dick Crouch, Jan Van Eijck, Chris Fox, Josef Van Genabith, Jan Jaspars, Hans Kamp, David Milward, Manfred Pinkal, Massimo Poesio, and Steve Pulman. Using the framework. Deliverable D16, The FRACAS Project, 1996.

Ido Dagan, Oren Glickman, and Bernardo Magnini. The pascal recognising textual entailment challenge. In Joaquin Quiñonero-Candela, Ido Dagan, Bernardo Magnini, and Florence d'Alché Buc, editors, *Machine Learning Challenges. Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Tectual Entailment*, pages 177–190, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg. ISBN 978-3-540-33428-6.

Florian Daniel, Pavel Kucherbaev, Cinzia Cappiello, Boualem Benatallah, and Mohammad Allahbakhsh. Quality control in crowdsourcing: A survey of quality attributes, assessment techniques, and assurance actions. *ACM Comput. Surv.*, 51(1), 2018. ISSN 0360-0300. doi: 10.1145/3148148. URL https://doi.org/10.1145/3148148.

A. Philip Dawid and Allan M. Skene. 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, 1979. ISSN 00359254, 14679876. URL http://www.jstor.org/stable/2346806.

Marie-Catherine de Marneffe and Christopher Potts. *Developing Linguistic Theories Using Annotated Corpora*, pages 411–438. Springer Netherlands, Dordrecht, 2017. ISBN 978-94-024-0881-2. doi: 10.1007/978-94-024-0881-2_16. URL https://doi.org/10.1007/978-94-024-0881-2_16.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423.

Anca Dumitrache. *Truth in Disagreement*. PhD thesis, Free University Amsterdam, 2019.

Anca Dumitrache, Lora Aroyo, and Chris Welty. Crowdsourcing semantic label propagation in relation classification. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 16–21, Brussels, Belgium, 2018a. Association for Computational Linguistics. doi: 10.18653/v1/W18-5503. URL https://aclanthology.org/W18-5503.

Anca Dumitrache, Lora Aroyo, and Chris Welty. Crowdsourcing ground truth for medical relation extraction. *ACM Trans. Interact. Intell. Syst.*, 8(2), 2018b. ISSN 2160-6455. doi: 10.1145/3152889. URL https://doi.org/10.1145/3152889.

Anca Dumitrache, Oana Inel, Lora Aroyo, Benjamin Timmermans, and Chris Welty. Crowdtruth 2.0: Quality metrics for crowdsourcing with disagreement (short paper). In Lora Aroyo, Anca Dumitrache, Praveen K. Paritosh, Alexander J. Quinn, Chris Welty, Alessandro Checco, Gianluca Demartini, Ujwal Gadiraju, and Cristina Sarasua, editors, *Proceedings of the 1st Workshop on Subjectivity, Ambiguity and Disagreement in Crowdsourcing, and Short Paper Proceedings of the 1st Workshop on Disentangling the Relation Between Crowdsourcing and Bias Management (SAD 2018 and CrowdBias 2018) co-located the 6th AAAI Conference on Human Computation and Crowdsourcing (HCOMP 2018), Zürich, Switzerland, July 5, 2018*, volume 2276 of *CEUR Workshop Proceedings*, pages 11–18. CEUR-WS.org, 2018c. URL http://ceur-ws.org/Vol-2276/paper2.pdf.

Anca Dumitrache, Oana Inel, Benjamin Timmermans, Carlos Martinez-Ortiz, Robert-Jan Sips, Lora Aroyo, and Chris Welty. Empirical methodology for crowdsourcing ground truth. *CoRR*, abs/1809.08888, 2018d. URL http://arxiv.org/abs/1809.08888.

Anca Dumitrache, Lora Aroyo, and Chris Welty. A crowdsourced frame disambiguation corpus with ambiguity. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2164–2170, Minneapolis, Minnesota, 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1224. URL https://aclanthology.org/N19-1224.

Paul Felt, Eric Ringger, Jordan Boyd-Graber, and Kevin Seppi. Making the most of crowdsourced document annotations: Confused supervised LDA. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning*, pages 194–203, Beijing, China, 2015. Association for Computational Linguistics. doi: 10.18653/v1/K15-1020. URL https://aclanthology.org/K15-1020.

Michael Firman, Neill D. F. Campbell, Lourdes Agapito, and Gabriel J. Brostow. Diversenet: When one right answer is not enough. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 5598–5607. IEEE Computer Society, 2018. doi: 10.1109/CVPR.2018.00587. URL http://openaccess.thecvf.com/content_cvpr_2018/html/Firman_DiverseNet_When_One_CVPR_2018_paper.html.

Joseph L. Fleiss, Bruce Levin, and Myunghee Cho Paik. *The Measurement of Interrater Agreement*, chapter 18, pages 598–626. John Wiley & Sons, Ltd, 2004. ISBN 9780471445425. doi: 10.1002/0471445428.ch18. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/0471445428.ch18.

Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. *CoRR*, abs/2010.01412, 2020. URL https://arxiv.org/abs/2010.01412.

Tommaso Fornaciari, Alexandra Uma, Silviu Paun, Barbara Plank, Dirk Hovy, and Massimo Poesio. 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*, pages 2591–2597, Online, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.204. URL https://aclanthology.org/2021.naacl-main.204.

Xavier Gastaldi. Shake-Shake regularization of 3-branch residual networks. In *5th International Conference on Learning Representations, ICLR*, 2016.

Aritra Ghosh, Himanshu Kumar, and P. S. Sastry. Robust loss functions under label noise for deep neural networks. In Satinder P. Singh and Shaul Markovitch, editors, *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, pages 1919–1925. AAAI Press, 2017. URL http://aaai.org/ocs/index.php/AAAI/AAAI17/paper/view/14759.

Kevin Gimpel, Nathan Schneider, Brendan O'Connor, Dipanjan Das, Daniel Mills, Jacob Eisenstein, Michael Heilman, Dani Yogatama, Jeffrey Flanigan, and Noah A. Smith. Part-of-speech tagging for Twitter: Annotation, features, and experiments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 42–47, Portland, Oregon, USA, 2011. Association for Computational Linguistics. URL https://aclanthology.org/P11-2008.

Benjamin Graham. Fractional max-pooling. *ArXiv*, abs/1412.6071, 2014.

Melody Guan, Varun Gulshan, Andrew Dai, and Geoffrey Hinton. Who said what: Modeling individual labelers improves classification. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1), Apr. 2018. URL https://ojs.aaai.org/index.php/AAAI/article/view/11756.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 1321–1330. PMLR, 2017. URL http://proceedings.mlr.press/v70/guo17a.html.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society, 2016. doi: 10.1109/CVPR.2016.90. URL https://doi.org/10.1109/CVPR.2016.90.

Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. In *NIPS Deep Learning and Representation Learning Workshop*, 2015. URL http://arxiv.org/abs/1503.02531.

Yufang Hou. Incremental fine-grained information status classification using attention-based LSTMs. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1880–1890, Osaka, Japan, 2016. The COLING 2016 Organizing Committee. URL https://aclanthology.org/C16-1177.

Yufang Hou, Katja Markert, and Michael Strube. Global inference for bridging anaphora resolution. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 907–917, Atlanta, Georgia, 2013. Association for Computational Linguistics. URL https://aclanthology.org/N13-1111.

Dirk Hovy, Taylor Berg-Kirkpatrick, Ashish Vaswani, and Eduard Hovy. 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, 2013. Association for Computational Linguistics. URL https://aclanthology.org/N13-1132.

Eduard Hovy, Mitchell Marcus, Martha Palmer, Lance Ramshaw, and Ralph Weischedel. OntoNotes: The 90% solution. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pages 57–60, New York City, USA, 2006. Association for Computational Linguistics. URL https://aclanthology.org/N06-2015.

Nancy Ide and James Pustejovsky, editors. *The Handbook of Linguistic Annotation.* Springer, 2017.

Oana Inel and Lora Aroyo. Harnessing diversity in crowds and machines for better NER performance. In Eva Blomqvist, Diana Maynard, Aldo Gangemi, Rinke Hoekstra, Pascal Hitzler, and Olaf Hartig, editors, *The Semantic Web - 14th International Conference, ESWC 2017, Portorož, Slovenia, May 28 - June 1, 2017, Proceedings, Part I*, volume 10249 of *Lecture Notes in Computer Science*, pages 289–304, 2017. doi: 10.1007/978-3-319-58068-5_18. URL https://doi.org/10.1007/978-3-319-58068-5_18.

Oana Inel, Khalid Khamkham, Tatiana Cristea, Anca Dumitrache, Arne Rutjes, Jelle Ploeg, Lukasz Romaszko, Lora Aroyo, and Robert-Jan Sips. Crowdtruth: Machine-human computation framework for harnessing disagreement in gathering annotated data. In *International Semantic Web Conference (ISWC)*, pages 486–504, 2014. doi: 10.1007/978-3-319-11915-1_31.

Emily Jamison and Iryna Gurevych. 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, Lisbon, Portugal, 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1035. URL https://aclanthology.org/D15-1035.

David Jurgens. Embracing ambiguity: A comparison of annotation methodologies for crowdsourcing word sense labels. In *Proceedings of the 2013 Conference of the North*

*American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 556–562, Atlanta, Georgia, 2013. Association for Computational Linguistics. URL `https://aclanthology.org/N13-1062`.

Ece Kamar, Ashish Kapoor, and Eric Horvitz. Identifying and accounting for task-dependent bias in crowdsourcing. In *Third AAAI Conference on Human Computation and Crowdsourcing*, 2015.

Kian Kenyon-Dean, Eisha Ahmed, Scott Fujimoto, Jeremy Georges-Filteau, Christopher Glasz, Barleen Kaur, Auguste Lalande, Shruti Bhanderi, Robert Belfer, Nirmal Kanagasabai, Roman Sarrazingendron, Rohit Verma, and Derek Ruths. Sentiment analysis: It's complicated! In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1886–1895, New Orleans, Louisiana, 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1171. URL `https://aclanthology.org/N18-1171`.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL `http://arxiv.org/abs/1412.6980`.

Olga Krasavina and Christian Chiarcos. PoCoS - Potsdam Coreference Scheme. In *Proceedings of the Linguistic Annotation Workshop*, pages 156–163, Prague, Czech Republic, 2007. Association for Computational Linguistics. URL `https://aclanthology.org/W07-1525`.

Solomon Kullback and Richard A Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951.

John P Lalor, Hao Wu, and Hong Yu. Soft label memorization-generalization for natural language inference. *arXiv preprint arXiv:1702.08563*, 2017. URL `https://arxiv.org/abs/1702.08563`.

Kenton Lee, Luheng He, and Luke Zettlemoyer. Higher-order coreference resolution with coarse-to-fine inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 687–692, New Orleans, Louisiana, 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-2108. URL `https://aclanthology.org/N18-2108`.

Yuan Li, Benjamin I. P. Rubinstein, and Trevor Cohn. Exploiting worker correlation for label aggregation in crowdsourcing. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 3886–3895. PMLR, 2019. URL `http://proceedings.mlr.press/v97/li19i.html`.

Jianhua Lin. Divergence Measures Based on the Shannon Entropy. *IEEE Transactions on Information Theory*, 1991. ISSN 15579654. doi: 10.1109/18.61115.

Andrey Malinin and Mark J. F. Gales. Predictive uncertainty estimation via prior networks. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 7047–7058, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/3ea2db50e62ceefceaf70a9d9a56a6f4-Abstract.html.

Christopher D. Manning. Part-of-speech tagging from 97% to 100%: Is it time for some linguistics? In Alexander F. Gelbukh, editor, *Computational Linguistics and Intelligent Text Processing*, pages 171–189, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg. ISBN 978-3-642-19400-9.

Katja Markert, Yufang Hou, and Michael Strube. Collective classification for fine-grained information status. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 795–804, Jeju Island, Korea, 2012. Association for Computational Linguistics. URL https://aclanthology.org/P12-1084.

Héctor Martínez Alonso, Barbara Plank, Arne Skjærholt, and Anders Søgaard. Learning to parse with IAA-weighted loss. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1357–1361, Denver, Colorado, 2015. Association for Computational Linguistics. doi: 10.3115/v1/N15-1152. URL https://aclanthology.org/N15-1152.

Héctor Martínez Alonso, Anders Johannsen, and Barbara Plank. Supersense tagging with inter-annotator disagreement. In *Proceedings of the 10th Linguistic Annotation Workshop held in conjunction with ACL 2016 (LAW-X 2016)*, pages 43–48, Berlin, Germany, 2016. Association for Computational Linguistics. doi: 10.18653/v1/W16-1706. URL https://aclanthology.org/W16-1706.

Tomás Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. In Christopher J. C. Burges, Léon Bottou, Zoubin Ghahramani, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pages 3111–3119, 2013. URL https://proceedings.neurips.cc/paper/2013/hash/9aa42b31882ec039965f3c4923ce901b-Abstract.html.

Mike Mintz, Steven Bills, Rion Snow, and Daniel Jurafsky. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of* *the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 1003–1011, Suntec, Singapore, 2009. Association for Computational Linguistics. URL https://aclanthology.org/P09-1113.

Volodymyr Mnih and Geoffrey E. Hinton. Learning to label aerial images from noisy data. In *Proceedings of the 29th International Conference on Machine Learning, ICML 2012, Edinburgh, Scotland, UK, June 26 - July 1, 2012*. icml.cc / Omnipress, 2012. URL http://icml.cc/2012/papers/318.pdf.

Pablo G. Moreno, Antonio Artes-Rodriguez, Yee Whye Teh, Fern, and o Perez-Cruz. Bayesian nonparametric crowdsourcing. *Journal of Machine Learning Research*, 16 (48):1607–1627, 2015. URL http://jmlr.org/papers/v16/moreno15a.html.

Anna Nedoluzhko, Michal Novák, Silvie Cinková, Marie Mikulová, and Jiří Mírovský. Coreference in Prague Czech-English Dependency Treebank. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 169–176, Portorož, Slovenia, 2016. European Language Resources Association (ELRA). URL https://aclanthology.org/L16-1026.

Malvina Nissim, Shipra Dingare, Jean Carletta, and Mark Steedman. An annotation scheme for information status in dialogue. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, Lisbon, Portugal, 2004. European Language Resources Association (ELRA). URL http://www.lrec-conf.org/proceedings/lrec2004/pdf/638.pdf.

Curtis Northcutt, Lu Jiang, and Isaac Chuang. Confident learning: Estimating uncertainty in dataset labels. *J. Artif. Int. Res.*, 70:1373–1411, 2021. ISSN 1076-9757. doi: 10.1613/jair.1.12125. URL https://doi.org/10.1613/jair.1.12125.

Emmanuel Osei-Brefo, Thanet Markchom, and Huizhi Liang. UOR at SemEval-2021 task 12: On crowd annotations; learning with disagreements to optimise crowd truth. In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 1303–1309, Online, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.semeval-1.186. URL https://aclanthology.org/2021.semeval-1.186.

Rebecca J. Passonneau and Bob Carpenter. The benefits of a model of annotation. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 187–195, Sofia, Bulgaria, 2013. Association for Computational Linguistics. URL https://aclanthology.org/W13-2323.

Rebecca J. Passonneau, Vikas Bhardwaj, Ansaf Salleb-Aouissi, and Nancy Ide. Multiplicity and word sense: evaluating and learning from multiply labeled word sense annotations. *Language Resources and Evaluation*, 46(2):219–252, 2012. doi: 10.1007/s10579-012-9188-x.

Silviu Paun, Bob Carpenter, Jon Chamberlain, Dirk Hovy, Udo Kruschwitz, and Massimo Poesio. Comparing Bayesian models of annotation. *Transactions of the Association for Computational Linguistics*, 6:571–585, 2018. doi: 10.1162/tacl_a_00040. URL `https://aclanthology.org/Q18-1040`.

Ellie Pavlick and Tom Kwiatkowski. Inherent disagreements in human textual inferences. *Transactions of the Association for Computational Linguistics*, 7:677–694, 2019. doi: 10.1162/tacl_a_00293. URL `https://aclanthology.org/Q19-1043`.

Karl Pearson. Mathematical contributions to the theory of evolution. iii. regression, heredity, and panmixia. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 187:253–318, 1896. ISSN 02643952. URL `http://www.jstor.org/stable/90707`.

Jeffrey Pennington, Richard Socher, and Christopher Manning. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1162. URL `https://aclanthology.org/D14-1162`.

Joshua C. Peterson, Ruairidh M. Battleday, Thomas L. Griffiths, and Olga Russakovsky. 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, 2019. doi: 10.1109/ICCV.2019.00971. URL `https://doi.org/10.1109/ICCV.2019.00971`.

Slav Petrov, Dipanjan Das, and Ryan McDonald. 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, 2012. European Language Resources Association (ELRA). URL `http://www.lrec-conf.org/proceedings/lrec2012/pdf/274_Paper.pdf`.

Barbara Plank, Dirk Hovy, and Anders Søgaard. 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, 2014a. Association for Computational Linguistics. doi: 10.3115/v1/E14-1078. URL `https://aclanthology.org/E14-1078`.

Barbara Plank, Dirk Hovy, and Anders Søgaard. Linguistically debatable or just plain wrong? In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 507–511, Baltimore, Maryland, 2014b. Association for Computational Linguistics. doi: 10.3115/v1/P14-2083. URL `https://aclanthology.org/P14-2083`.

Barbara Plank, Anders Søgaard, and Yoav Goldberg. 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, 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-2067. URL https://aclanthology.org/P16-2067.

John C. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *ADVANCES IN LARGE MARGIN CLASSIFIERS*, pages 61–74. MIT Press, 1999.

Massimo Poesio and Ron Artstein. The reliability of anaphoric annotation, reconsidered: Taking ambiguity into account. In *Proceedings of the Workshop on Frontiers in Corpus Annotations II: Pie in the Sky*, pages 76–83, Ann Arbor, Michigan, 2005. Association for Computational Linguistics. URL https://aclanthology.org/W05-0311.

Massimo Poesio and Ron Artstein. Anaphoric annotation in the ARRAU corpus. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco, 2008. European Language Resources Association (ELRA). URL http://www.lrec-conf.org/proceedings/lrec2008/pdf/297_paper.pdf.

Massimo Poesio, Patrick Sturt, Ron Artstein, and Ruth Filik. Underspecification and anaphora: Theoretical issues and preliminary evidence. *Discourse processes*, 42(2): 157–175, 2006.

Massimo Poesio, Uwe Reyle, and Rosemary Stevenson. *Justified Sloppiness In Anaphoric Reference*, pages 11–31. Springer Netherlands, Dordrecht, 2007. ISBN 978-1-4020-5958-2. doi: 10.1007/978-1-4020-5958-2_2. URL https://doi.org/10.1007/978-1-4020-5958-2_2.

Massimo Poesio, Jon Chamberlain, Udo Kruschwitz, Livio Robaldo, and Luca Ducceschi. Phrase detectives: Utilizing collective intelligence for internet-scale language resource creation. *ACM Transactions on Intelligent Interactive Systems*, 3(1), 2013. doi: 10.1145/2448116.2448119. URL http://dl.acm.org/citation.cfm?id=2448119.

Massimo Poesio, Jon Chamberlain, and Udo Kruschwitz. Crowdsourcing. In N. Ide and J. Pustejovsky, editors, *The Handbook of Linguistic Annotation*, pages 277–295. Springer, 2017.

Massimo Poesio, Yulia Grishina, Varada Kolhatkar, Nafise Moosavi, Ina Roesiger, Adam Roussel, Fabian Simonjetz, Alexandra Uma, Olga Uryupina, Juntao Yu, and Heike Zinsmeister. Anaphora resolution with the ARRAU corpus. In *Proceedings of the First Workshop on Computational Models of Reference, Anaphora and Coreference*, pages 11–22, New Orleans, Louisiana, 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-0702. URL https://aclanthology.org/W18-0702.

Massimo Poesio, Jon Chamberlain, Silviu Paun, Juntao Yu, Alexandra Uma, and Udo Kruschwitz. A crowdsourced corpus of multiple judgments and disagreement on anaphoric interpretation. In *Proceedings of the 2019 Conference of the North*

*American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1778–1789, Minneapolis, Minnesota, 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1176. URL `https://aclanthology.org/N19-1176`.

Sameer Pradhan, Lance Ramshaw, Mitchell Marcus, Martha Palmer, Ralph Weischedel, and Nianwen Xue. CoNLL-2011 shared task: Modeling unrestricted coreference in OntoNotes. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–27, Portland, Oregon, USA, 2011. Association for Computational Linguistics. URL `https://aclanthology.org/W11-1901`.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Olga Uryupina, and Yuchen Zhang. CoNLL-2012 shared task: Modeling multilingual unrestricted coreference in OntoNotes. In *Joint Conference on EMNLP and CoNLL - Shared Task*, pages 1–40, Jeju Island, Korea, 2012. Association for Computational Linguistics. URL `https://aclanthology.org/W12-4501`.

Ellen F. Prince. Toward a taxonomy of given-new information. In P. Cole, editor, *Radical Pragmatics*, pages 223–256. Academic Press, New York, 1981.

Ellen F. Prince. The ZPG letter: subjects, definiteness, and information status. In S. Thompson and W. Mann, editors, *Discourse description: diverse analyses of a fund-raising text*, pages 295–325. John Benjamins, 1992.

Nguyen Quoc Viet Hung, Nguyen Thanh Tam, Lam Ngoc Tran, and Karl Aberer. An evaluation of aggregation techniques in crowdsourcing. In Xuemin Lin, Yannis Manolopoulos, Divesh Srivastava, and Guangyan Huang, editors, *Web Information Systems Engineering – WISE 2013*, pages 1–15, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg. ISBN 978-3-642-41154-0.

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

Marta Recasens, Ed Hovy, and M. Antonia Martí. Identity, non-identity, and near-identity: Addressing the complexity of coreference. *Lingua*, 121(6):1138–1152, 2011.

Dennis Reidsma and Jean Carletta. Reliability measurement without limits. *Computational Linguistics*, 34(3):319–326, 2008. ISSN 0891-2017. doi: 10.1162/coli.2008.34.3.319. URL `https://doi.org/10.1162/coli.2008.34.3.319`.

Dennis Reidsma and Rieks op den Akker. Exploiting 'subjective' annotations. In *Coling 2008: Proceedings of the workshop on Human Judgements in Computational Linguistics*, pages 8–16, Manchester, UK, 2008. Coling 2008 Organizing Committee. URL `https://aclanthology.org/W08-1203`.

Arndt Riester, David Lorenz, and Nina Seemann. A recursive annotation scheme for referential information status. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta, 2010. European Language Resources Association (ELRA). URL `http://www.lrec-conf.org/proceedings/lrec2010/pdf/764_Paper.pdf`.

Filipe Rodrigues and Francisco C. Pereira. Deep learning from crowds. In Sheila A. McIlraith and Kilian Q. Weinberger, editors, *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 1611–1618. AAAI Press, 2018. URL `https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16102`.

Filipe Rodrigues, Mariana Lourenco, Bernardete Ribeiro, and Francisco Pereira. Learning supervised topic models for classification and regression from crowds. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP:1–1, 2017. doi: 10.1109/TPAMI.2017.2648786.

Willibald Ruch. Psychology of humor. In Victor Raskin, editor, *The Primer of Humor Research*, number 8 in Humor Research, pages 17–100. Mouton de Gruyter, Berlin, 2008. ISBN 978-3-11-018616-1. doi: 10.1515/9783110198492.17.

Sebastian Ruder. An overview of multi-task learning in deep neural networks, 2017.

Bryan Russell, Antonio Torralba, Kevin Murphy, and William Freeman. Labelme: A database and web-based tool for image annotation. *International Journal of Computer Vision*, 77, 2008. doi: 10.1007/s11263-007-0090-8.

Viktoriia Sharmanska, Daniel Hernández-Lobato, José Miguel Hernández-Lobato, and Novi Quadrianto. Ambiguity helps: Classification with disagreements in crowdsourced annotations. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 2194–2202. IEEE Computer Society, 2016. doi: 10.1109/CVPR.2016.241. URL `https://doi.org/10.1109/CVPR.2016.241`.

Victor S. Sheng, Foster Provost, and Panagiotis G. Ipeirotis. 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*, KDD '08, page 614–622, New York, NY, USA, 2008. Association for Computing Machinery. ISBN 9781605581934. doi: 10.1145/1401890.1401965. URL `https://doi.org/10.1145/1401890.1401965`.

Aashish Sheshadri and Matthew Lease. Square: A benchmark for research on computing crowd consensus. *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, 1(1):156–164, Nov. 2013. URL `https://ojs.aaai.org/index.php/HCOMP/article/view/13088`.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *CoRR*, abs/1312.6034, 2013.

Edwin Simpson and Iryna Gurevych. A Bayesian approach for sequence tagging with crowds. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1093–1104, Hong Kong, China, 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1101. URL https://aclanthology.org/D19-1101.

Edwin Simpson and Iryna Gurevych. Scalable Bayesian preference learning for crowds. *Machine Learning*, 109(4):689–718, 2020. doi: 10.1007/s10994-019-05867-2.

Edwin Simpson, Erik-Lân Do Dinh, Tristan Miller, and Iryna Gurevych. Predicting humorousness and metaphor novelty with Gaussian process preference learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5716–5728, Florence, Italy, 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1572. URL https://aclanthology.org/P19-1572.

Padhraic Smyth, Usama Fayyad, Michael Burl, Pietro Perona, and Pierre Baldi. Inferring ground truth from subjective labelling of venus images. In *Proceedings of the 7th International Conference on Neural Information Processing Systems*, NIPS'94, page 1085–1092, Cambridge, MA, USA, 1994. MIT Press.

Rion Snow, Brendan O Connor, Daniel Jurafsky, Andrew Y Ng, Dolores Labs, and Capp St. Cheap and fast - but is it good? Evaluation non-expert annotiations for natural language tasks. *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 254–263, 2008. doi: 10.1.1.142.8286. URL http://www.aclweb.org/anthology/D08-1027.

Anders Søgaard, Anders Johannsen, Barbara Plank, Dirk Hovy, and Hector Martínez Alonso. 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, 2014. Association for Computational Linguistics. doi: 10.3115/v1/W14-1601. URL https://aclanthology.org/W14-1601.

Kihyuk Sohn. Improved deep metric learning with multi-class n-pair loss objective. In Daniel D. Lee, Masashi Sugiyama, Ulrike von Luxburg, Isabelle Guyon, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 1849–1857, 2016. URL https://proceedings.neurips.cc/paper/2016/hash/6b180037abbebea991d8b1232f8a8ca9-Abstract.html.

Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin A. Riedmiller. Striving for simplicity: The all convolutional net. *CoRR*, abs/1412.6806, 2015.

Manfred Stede. Disambiguating rhetorical structure. *Research in Language and Computation*, 6(3–4):311–332, 2008.

Georg Stemmer, Stefan Steidl, Elmar Nöth, Heinrich Niemann, and Anton Batliner. Comparison and combination of confidence measures. In Petr Sojka, Ivan Kopeček, and Karel Pala, editors, *Text, Speech and Dialogue*, pages 181–188, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg. ISBN 978-3-540-46154-8.

Dietrich Trautmann, Johannes Daxenberger, Christian Stab, Hinrich Schütze, and Iryna Gurevych. Robust argument unit recognition and classification. *CoRR*, abs/1904.09688, 2019. URL `http://arxiv.org/abs/1904.09688`.

Alexandra Uma, Tommaso Fornaciari, Dirk Hovy, Silviu Paun, Barbara Plank, and Massimo Poesio. A case for soft loss functions. In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, volume 8, pages 173–177, 2020.

Alexandra Uma, Tommaso Fornaciari, Anca Dumitrache, Tristan Miller, Jon Chamberlain, Barbara Plank, Edwin Simpson, and Massimo Poesio. SemEval-2021 task 12: Learning with disagreements. In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 338–347, Online, 2021a. Association for Computational Linguistics. doi: 10.18653/v1/2021.semeval-1.41. URL `https://aclanthology.org/2021.semeval-1.41`.

Alexandra Uma, Tommaso Fornaciari, Dirk Hovy, Silviu Paun, Barbara Plank, and Massimo Poesio. Learning with disagreements. *Journal of Artificial Intelligence Research*, 4(2):201–213, 2021b. An optional note.

Olga Uryupina, Ron Artstein, Antonella Bristot, Federica Cavicchio, Francesca Delogu, Kepa Rodriguez, and Massimo Poesio. Annotating a broad range of anaphoric phenomena, in a variety of genres: the ARRAU corpus. *Journal of Natural Language Engineering*, 26(1), 2020.

Yannick Versley. Vagueness and referential ambiguity in a large-scale annotated corpus. *Research on Language and Computation*, 6:333–353, 2008.

Luis von Ahn and Laura Dabbish. Designing games with a purpose. *Communications of the ACM*, 51(8):58–67, 2008. ISSN 0001-0782. doi: 10.1145/1378704.1378719. URL `https://doi.org/10.1145/1378704.1378719`.

Chang Wang and James Fan. Medical relation extraction with manifold models. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 828–838, Baltimore, Maryland, 2014. Association for Computational Linguistics. doi: 10.3115/v1/P14-1078. URL `https://aclanthology.org/P14-1078`.

Zeerak Waseem. Are you a racist or am I seeing things? annotator influence on hate speech detection on Twitter. In *Proceedings of the First Workshop on NLP* *and Computational Social Science*, pages 138–142, Austin, Texas, 2016. Association for Computational Linguistics. doi: 10.18653/v1/W16-5618. URL https://aclanthology.org/W16-5618.

Jacob Whitehill, Paul Ruvolo, Tingfan Wu, Jacob Bergsma, and Javier R. Movellan. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. In Yoshua Bengio, Dale Schuurmans, John D. Lafferty, Christopher K. I. Williams, and Aron Culotta, editors, *Advances in Neural Information Processing Systems 22: 23rd Annual Conference on Neural Information Processing Systems 2009. Proceedings of a meeting held 7-10 December 2009, Vancouver, British Columbia, Canada*, pages 2035–2043. Curran Associates, Inc., 2009. URL https://proceedings.neurips.cc/paper/2009/hash/f899139df5e1059396431415e770c6dd-Abstract.html.

Hui Yang, Anne De Roeck, Vincenzo Gervasi, Alistair Willis, and Bashar Nuseibeh. Analysing anaphoric ambiguity in natural language requirements. *Requirements Engineering*, 16:163–189, 2011. doi: 10.1007/s00766-011-0119-y.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489, San Diego, California, 2016. Association for Computational Linguistics. doi: 10.18653/v1/N16-1174. URL https://aclanthology.org/N16-1174.

Juntao Yu, Alexandra Uma, and Massimo Poesio. A cluster ranking model for full anaphora resolution. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 11–20, Marseille, France, 2020. European Language Resources Association. ISBN 979-10-95546-34-4. URL https://aclanthology.org/2020.lrec-1.2.

Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In Richard C. Wilson, Edwin R. Hancock, and William A. P. Smith, editors, *Proceedings of the British Machine Vision Conference 2016, BMVC 2016, York, UK, September 19-22, 2016*. BMVA Press, 2016. URL http://www.bmva.org/bmvc/2016/papers/paper087/index.html.

Jing Zhang, Xindong Wu, and Victor Sheng. Learning from crowdsourced labeled data: a survey. *Artificial Intelligence Review*, 2016. doi: 10.1007/s10462-016-9491-9.

Yudian Zheng, Guoliang Li, Yuanbing Li, Caihua Shan, and Reynold Cheng. Truth inference in crowdsourcing: Is the problem solved? *Proceedings of the VLDB Endowment*, 10(5):541–552, 2017.