# More Labels or Cases?
# Assessing Label Variation in Natural Language Inference

**Cornelia Gruber**[*1♠] **Katharina Hechinger**[*1♠] **Matthias Aßenmacher**[1,2♠]
**Göran Kauermann**[1♠] **Barbara Plank**[2,3♣]
[1] Department of Statistics, LMU Munich, Germany
[2] Munich Center for Machine Learning (MCML), Munich, Germany
[3] Center for Information and Language Processing (CIS), LMU Munich, Germany
[♠]{cornelia.gruber, katharina.hechinger, matthias, goeran.kauermann}@stat.uni-muenchen.de
[♣]bplank@cis.uni-muenchen.de

## Abstract

In this work, we analyze the uncertainty that is inherently present in the labels used for supervised machine learning in natural language inference (NLI). In cases where multiple annotations per instance are available, neither the majority vote nor the frequency of individual class votes is a trustworthy representation of the labeling uncertainty. We propose modeling the votes via a Bayesian mixture model to recover the data-generating process, i.e., the posterior distribution of the "true" latent classes, and thus gain insight into the class variations. This will enable a better understanding of the confusion happening during the annotation process. We also assess the stability of the proposed estimation procedure by systematically varying the numbers of i) instances and ii) labels. Thereby, we observe that few instances with many labels can predict the latent class borders reasonably well, while the estimation fails for many instances with only a few labels. This leads us to conclude that multiple labels are a crucial building block for properly analyzing label uncertainty.

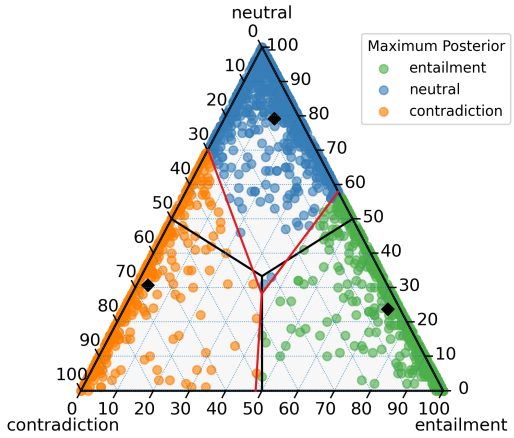



Figure 1: Scatter plot of the vote distribution of ChaosSNLI. Each point represents one instance. Its location is determined by the vote distribution. Corner points represent 100 votes for the respective class, i.e., *entailment*, *neutral*, *contradiction* for the bottom right, top, and bottom left, respectively. Solid black lines represent the border of class membership by majority vote. The color of the points is determined by the estimated latent class given by our model. Black diamonds describe the center points of the latent classes. Solid red lines represent the borders of latent class membership.

## 1 Introduction

Commonly, binary or multi-class classification settings in machine learning assume that a single gold label—representing the "true" class of an instance—can easily be acquired via human annotation. However, there are numerous examples where remarkable variations between different annotators exist, challenging the validity of this assumption (Uma et al., 2021). This issue is especially prevalent in datasets relating to the difficult task of perceiving human language, such as natural language inference (NLI). In NLI, the textual entailment of two sentences is to be determined. There exists an increasing body of work documenting inherent disagreement in labeling for NLI (Pavlick and Kwiatkowski, 2019; Nie et al., 2020; Zhang and de Marneffe, 2021; Jiang et al., 2023). Such human label variation can be caused by context dependency and subjectivity, amongst others, and is ubiquitous (Plank, 2022). Moreover, human label variation is different from annotation errors, as plausible, linguistic reasons for such variation exist (Jiang and de Marneffe, 2022).

To provide new grounds to study human variation in labeling, Nie et al. (2020) collected the ChaosNLI (Collective HumAn OpinionS on Natural Language Inference) dataset. ChaosNLI comprises 100 labels per instance from quality-controlled annotators for each of the ambiguous



*Proceedings of the Third Workshop on Understanding Implicit and Underspecified Language*, pages 22–32

instances from multiple NLI-related datasets. In this paper, we analyze ChaosSNLI, a sub-dataset of ChaosNLI based on the Stanford Natural Language Inference (SNLI) data (Bowman et al., 2015). Several works on NLI (Pavlick and Kwiatkowski, 2019; Nie et al., 2020) show that many instances exhibit high human disagreement or uncertainty, i.e., human labelers do not agree on a single class, resulting in a high spread of the annotators' votes among multiple classes. Less work has looked at label variation and stability from a data-generating process viewpoint in light of uncertainty.

Uncertainty in machine learning and NLP is, however, gaining increased attention recently (Hüllermeier and Waegeman, 2021; Gruber et al., 2023; Baan et al., 2023). Different lines of research study sources of uncertainty in various parts of machine learning, such as the data itself, the model choice, the estimation procedure, and model deployment (Gruber et al., 2023). Early works characterize uncertainty in terms of reducible and irreducible randomness (Hüllermeier and Waegeman, 2021), while some works argue that this line is fuzzy (Gruber et al., 2023; Baan et al., 2023).

Variation in labels is part of the uncertainty in the data and is ubiquitous given the inherent ambiguity of language (Zhang et al., 2021). Yet, understanding the uncertainty in labels enables us to not only empirically investigate human confusion in annotated data, but also to gain insights on the classification task itself. For example, the complexity of detecting certain classes or the composition of class structures can be derived from voting patterns—this information can provide useful insights into task characteristics.

Therefore, in order to analyze the uncertainty in the label vote distribution of ChaosSNLI, we model the data-generating process and analyze the stability of the resulting estimation. To do so, we employ a Bayesian mixture model and recover the latent "true" class label, see also Hechinger et al. (2024). More precisely, we obtain the posterior probability for each of the classes and can thus assess the certainty for the class labels given the votes.

Our results could further be incorporated into a machine learning pipeline, e.g., by fitting a model on our latent classes instead of majority vote classes or class frequencies. This is, however, beyond the scope of this paper. In this work, we focus on the fundamental step of quantifying labeling uncertainty instead. We propose an estimation procedure and analyze its *stability* for different amounts of i) instances and ii) labels. Our work shows that more labels are more beneficial for stable estimation of uncertainty, while only a few instances already suffice. We also suggest new tools for *visual assessment* of the uncertainty in labels for three-way classification tasks (see Fig. 1).

**Contributions** With this paper, we contribute to a better understanding of label variation via a deep assessment of trustworthiness by 1) quantifying labeling uncertainty with Bayesian mixture models, 2) providing a novel visual tool for a better assessment of labeling uncertainty, and 3) deriving practical guidance for labeling tasks. We identify the benefit of using fewer cases with many labels rather than the other way around.[1]

## 2 Related Work

The need to analyze diverse human opinions in natural language inference is discussed by works including Pavlick and Kwiatkowski (2019) and Nie et al. (2020). Nie et al. (2020) show that some state-of-the-art models (including BERT, RoBERTa, XLNET, AL- BERT, DistilBERT, and BART) are neither designed nor able to capture human variation in labels and are therefore not appropriate. Their work also states that predicting the majority vote and predicting the human label distribution are distinct and seemingly conflicting objectives. In their benchmark study, all considered models performed consistently worse on examples with low human agreement. This indicates that analyzing label variation is of significant relevance for a more complete understanding of natural language inference.

Hovy et al. (2013) already advocated that majority voting might be the simplest but not most appropriate strategy for finding the correct label and, that modeling the votes leads to improved predicted label accuracy. The authors propose a method to separately model annotations from spamming and non-spamming annotators. Our methods differ in the way variation in labels is modeled. Hovy et al. (2013) explicitly model the behavior of annotators and assumes non-spamming annotators always provide the correct label, while votes by spamming annotators are drawn from a multinomial distribution. In contrast, our approach models human confusion in the annotation process, assuming equal levels of annotation skills. This is a reasonable assumption

[1]Code and data available at: https://github.com/corneliagru/label-variation-nli

for ChaosSNLI as all annotators undergo strict quality control, see Nie et al. (2020) for details. Nevertheless, both methods share the goal of estimating the distribution of the data-generating process and its parameters via an expectation-maximization (EM) algorithm.

Paun et al. (2018) compare various Bayesian approaches for modeling annotation. Based on their taxonomy, we employ a pooled model, i.e., assuming equal quality of the annotators. They conclude that such pooled models underperform, as the assumption that all annotators share the same ability is inappropriate in typical crowdsourcing settings. However, when information on individual annotators is unavailable, as is the case for the investigated ChaosSNLI dataset, pooling is inevitable.

The benefits of harnessing multiple labels are presented in Zhang et al. (2021). They demonstrate that improvements in accuracy can be achieved by varying the number of annotations for some examples within a given annotation budget. Our findings show a more nuanced picture supporting their claims, as we show the necessity of multiple annotations but a flattening value curve (see section 5).

## 3 Dataset and Problem Setting

We examine label uncertainty in NLI, a task for which textual entailment of two sentences is typically classified as either *entailment*, *neutral*, or *contradiction*. In ChaosSNLI (Nie et al., 2020), *multiple* annotations for *each* instance are provided. Example sentences of ChaosSNLI with their respective votes are shown in Table 1. Since those annotators do not necessarily agree with each other, we face a high degree of (human) label uncertainty. We chose this dataset as it provides a unique ground to explore label variation. Having access to a high amount of labels per instance is particularly valuable, but unfortunately not a common setting.

Our analysis is based on $N = 1,514$ instances with $J = 100$ labels, each, that originate from the development set of the SNLI dataset (Bowman et al., 2015). The original SNLI development set was generated by a multistep procedure, where first an initial annotator provides a text description of an image, i.e., generating the *premise*. Second, a different annotator constructs three *hypotheses* as an entailing, neutral, and contradicting description of the premise. Third, four more annotators, independent of the first two steps, provide labels for the premise-hypothesis pairs, i.e., classify the pairs into *entailment, neutral* or *contradiction*. This procedure yields five annotations per instance in total. In ChaosSNLI, examples, where only three out of those five annotators agree, are then relabeled by 100 quality-controlled annotators. For details on the quality control procedure, we refer to Nie et al. (2020). This relabeling procedure leads to a dataset, where instances with a high degree of uncertainty are overrepresented. Such a biased sample is valuable, as our main interest lies in understanding exactly those uncertain and hard-to-classify cases.

In the dataset, we observe that the most common class according to majority voting is *neutral*, with 53.7% of all examples, while *entailment* and *contradiction* amount to 27.8% and 18.5%, respectively. This already suggests that identifying *neutral* seems to be more challenging than discerning the other classes, as human annotators do not agree on those especially challenging examples that were collected for ChaosSNLI.

To gain a better understanding of label uncertainty in NLI, we analyze the annotations for the premise-hypothesis pairs available in ChaosSNLI. In order to detect hidden structures and comprehend label variation, we follow a statistical approach for modeling the label distribution. It is thus distinct from classical machine learning, where models are optimized for predictive power. However, our approach can ultimately be incorporated as a preprocessing step for predictive models. A precise description of our methodology can be found in section 4.

## 4 Modeling Approach

The main goal of this work is to explore the uncertainty inherent in the (multiple) labels of the sentence pairs in ChaosSNLI which is expressed by the distribution of the annotations. In order to formally describe the dataset with its multiple annotations and to assess label uncertainty, we use tools from statistical modeling. The multinomial mixture model provides the possibility to put multiple annotations into a distributional framework and subsequently estimate the associated parameters. Based on these parameters, a latent ground truth label can be derived for each instance, incorporating the uncertainty expressed by the distributions of the annotations over all instances. We follow the methodology proposed in Hechinger et al. (2024) for modeling multiple annotations via a Bayesian

| Context/Premise | Statement/Hypothesis | [E, N, C] |
|---|---|---|
| A boy in an orange shirt sells fruit from a street cart. | A boy is a street vendor. | [90, 10, 0] |
| A woman wearing a red hat and black coat. | The woman is asleep. | [0, 87, 13] |
| People walk amongst a traffic jam in a crowded city. | The cars are zooming past the people. | [3, 15, 82] |
| A woman holding a child in a purple shirt. | The woman is asleep at home. | [1, 53, 46] |

Table 1: Examples of ChaosSNLI. Annotators answered the question: "Given a context, a statement can be either: definitely correct (Entailment); or definitely incorrect (Contradiction); or neither (Neutral). Your goal is to choose the correct category for a given pair of context and statement."

mixture model.

First, let us introduce a formal description of the data. Each instance is a pair of $(X^{(i)}, \boldsymbol{Y}^{(i)}), i = 1, \ldots, N$, where $X^{(i)}$ denotes the sentence pair of premise and hypothesis and $\boldsymbol{Y}^{(i)}$ denotes the corresponding vote distribution. For this work, our focus lies on the latter exclusively, i.e., we only consider the vector of annotations for each instance. To explicitly represent votes for $K$ possible classes by $J$ different annotators, $\boldsymbol{Y}^{(i)}$ is set to $\boldsymbol{Y}^{(i)} = (Y_1^{(i)}, \ldots, Y_K^{(i)})$ with $Y_k^{(i)} = \sum_{j=1}^{J} \mathbb{1}(V_j^{(i)} = k)$. Here, $V_j^{(i)}$ denotes the individual vote for instance $i$ by annotator $j$. In ChaosSNLI we do not have access to individual annotator-specific votes, but observe $\boldsymbol{Y}^{(i)}$ directly. As mentioned above, we model the uncertainty inherent in the labels, so we omit $X^{(i)}$ and only analyze $\boldsymbol{Y}^{(i)}$. It is worth mentioning that incorporating the actual text is still possible for downstream tasks, but is out of the scope of this work.

In order to make use of the multinomial mixture model, we assume that each instance is associated with one true label, i.e., there exists an unambiguous ground truth. However, due to the inherent uncertainty in the perception of language, annotators are not easily capable of recovering the ground truth and they might vote for different classes. We denote the latent ground truth of each instance $X^{(i)}$ with $Z^{(i)} \in \{1, \ldots, K\}$. Again, to match our notation with the definition of a multivariate variable, we define $\boldsymbol{Z}^{(i)}$ as a one-hot encoded vector indicating the latent class, i.e., $\boldsymbol{Z}^{(i)} = (\mathbb{1}\{Z^{(i)} = 1\}, \ldots, \mathbb{1}\{Z^{(i)} = K\})$.

In the context of this particular dataset, as described in section 3, there exists a clearly defined ground truth that annotators should recover. This is due to the fact, that the annotator had one specific class in mind while inventing the hypothesis. Thus, the assumption of exactly one underlying "true" label is justified. However, this methodology can be applied beyond scenarios with known ground truth. In cases where no such information is available, the distributions of votes can serve as a valuable tool for deducing the latent labels.

**Model Framework** Let us now proceed to the analysis of the voting distribution $\boldsymbol{Y}^{(i)}$, which carries information about the latent true labels. We employ the following Bayesian modeling framework. First, considering the ground truth labels to be unobserved (or unobservable), they are assumed to follow a multinomial distribution

$$\boldsymbol{Z}^{(i)} \sim Multi(\boldsymbol{\pi}, 1) \text{ i.i.d.},$$

where $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_K)$ denote the prior probabilities for all classes. This distribution is also called the *prior* distribution. Given the true classes, the annotations are also assumed to be distributed multinomially, i.e.,

$$\boldsymbol{Y}^{(i)} | Z^{(i)} \sim Multi(\boldsymbol{\theta}_p, J). \quad (1)$$

This multinomial distribution describes the data *likelihood* conditional on $Z$. Here, the parameter vector $\boldsymbol{\theta}_p$ depends on the latent true class $Z^{(i)}$, i.e., the multinomial probabilities vary based on what we consider to be the true label. Hence, this parameter describes the probability of voting for a class given the true label. We can summarize the multinomial probability vectors of each latent, true class into a matrix $\boldsymbol{\Theta} = (\theta_{pk}, p, k = 1, \ldots, K)$, which can be interpreted as a confusion matrix. Formally, $\theta_{pk}$ describes the probability of an annotator voting for class $k$ given the instance has the true class $p$, i.e., using the notation in Eq. (1) we have $\boldsymbol{\theta}_p = (\theta_{p1}, \theta_{p2}, \ldots, \theta_{pK})$.

The key component of the model is the *posterior* distribution, i.e., the probabilities for an instance to truly belong to each of the classes given the observed annotations. These probabilities are cal-

culated as

$$\begin{aligned}\tau_p^{(i)} &= P(Z^{(i)} = p|\boldsymbol{Y}^{(i)}; \boldsymbol{\pi}, \boldsymbol{\Theta}) \\ &= \frac{P(Z^{(i)} = p; \boldsymbol{\pi}) P(\boldsymbol{Y}^{(i)}|Z^{(i)} = p; \boldsymbol{\Theta})}{P(\boldsymbol{Y}^{(i)}; \boldsymbol{\pi}, \boldsymbol{\Theta})} \\ &= \frac{\pi_p P(\boldsymbol{Y}^{(i)}; \boldsymbol{\theta}_p)}{\sum_{p'=1}^{K} \pi_{p'} P(\boldsymbol{Y}^{(i)}; \boldsymbol{\theta}_{p'})}.\end{aligned}$$

The class with the maximal posterior serves as an estimate for the latent ground truth, it is however also possible to use $\boldsymbol{\tau}$ in downstream tasks directly, i.e., for training a classifier on the probabilities instead of discrete class labels and thus directly incorporate the label uncertainty.

It is important to note that the prior modeling assumption of a single ground truth does not dictate the reality to be discrete, much more it enables us to compute the posterior distribution and quantify the evidence for each class, given the vote distribution. It thus allows us to model settings with ambiguous labels.

**Estimation Procedure** The model above includes unknown parameters, which we suggest estimating through maximum likelihood. As we are in the latent variable framework, straightforward estimation of the model parameters via maximum likelihood is, however, not possible. Instead, we apply an iterative estimation procedure to obtain parameter estimates. With the help of the expectation-maximization (EM) algorithm as introduced by Dempster et al. (1977), we can replace the latent class label $\boldsymbol{Z}^{(i)}$ with its expectation for each voting distribution. The expected latent class is thereby calculated given the data and the current parameter estimates and can be used afterward to update the estimates, leading to an iterative procedure that is performed until convergence. The algorithm can be outlined as follows, with additional details available in Hechinger et al. (2024) and in Appendix A. For the current parameter values at estimation iteration $(t)$, $\boldsymbol{\Theta} = \boldsymbol{\Theta}_{(t)}$ and $\boldsymbol{\pi} = \boldsymbol{\pi}_{(t)}$, one iterates over the two steps:

1. **E-Step:** Calculate the expectation of the full data likelihood given the data and the current estimates. Applying Bayes' rule, this simplifies to the computation of the expected latent class, given by posterior probabilities $\tau_p^{(i)}, i = 1, ..., N$ and $p = 1, ..., K$.

2. **M-Step:** Update the parameters $\boldsymbol{\Theta} = \boldsymbol{\Theta}_{(t+1)}$ and $\boldsymbol{\pi} = \boldsymbol{\pi}_{(t+1)}$ based on the posterior $\boldsymbol{\tau}$.

The final estimates are denoted as $\hat{\boldsymbol{\Theta}}$ and $\hat{\boldsymbol{\pi}}$. Our modeling approach harnesses the information retrieved from the annotations from all instances, as in every EM-step all instances are used for recalculating the estimates. This enables our method to incorporate knowledge about all annotation uncertainties and provide a comprehensive and holistic view of label variation.

**Label Switching** The classes obtained through mixture models are subject to label switching, i.e., their numbering is arbitrary and does not correspond to the original order anymore. This is a common issue in mixture models and can be resolved in various ways depending on the specific application at hand, as outlined by Stephens (2000). In this case, we apply a simple heuristic permutation to the latent classes. The original classes *entailment, contradiction, neutral*, denoted with index $k = 1, 2, 3$, are assigned to the respective latent classes $p = 1, 2, 3$ based on the diagonal entries of the estimated confusion matrix $\boldsymbol{\Theta}$. E.g., the class *entailment* is assigned to the mixture component, where the highest voting probability is *entailment*. This corresponds to the permutation $\sigma^{-1}(p) = \arg\max_k(\hat{\boldsymbol{\theta}}_p)$ and the latent classes are re-ordered accordingly.

To summarize, by allowing for human uncertainty, i.e., human confusion while labeling a certain instance, we can recover information on a latent class $Z$. The posterior distribution of the latent class is then a more trustworthy representation of the "true" class an instance belongs to, since all information contained in the full dataset is used for estimation, and not only the specific label distribution.

## 5 Results

### 5.1 Introspection by Visualization

As described earlier, the dataset ChaosSNLI (Nie et al., 2020) consists of $J = 100$ annotations for $K = 3$ classes. We propose to analyze human label variation in NLI with a novel visualization tool, to help gain insights into labeling. Figure 1 illustrates the distribution of votes present in ChaosSNLI, which we then contrast to the majority vote and our model's estimated class membership votes.

Each point in Figure 1 represents one instance, where its location is determined by the empirical distribution of votes. It is clearly visible by the density of dots that most instances cluster around the top of the plot, i.e., with many votes for *neutral*.

This is consistent with the distribution of majority votes (with *neutral* being observed 53.7% of times, as discussed in section 3). Furthermore, we observe that there is little confusion between *contradiction* and *entailment*, as almost no points lie close to the lower horizontal line or the vertical line starting in the center. This observation is intuitively plausible, due to the contrasting nature of the two labels of *entailment* vs. *contradiction*. Interestingly, this visualization tool helps us to quickly identify that there are cases in the datasets where many labels for both *entailment* and *contradiction* were observed.

In order to analyze our modeling result in relation to majority voting, we examine the borders between the three classes. Figure 1 shows the borders of the majority voting as solid black lines, which connect the center points of the axes, i.e., 50:50 votes for two of the classes, to the center, i.e., 33.33 votes for all three classes.

The borders between the latent classes are shown as red lines. To calculate these borders between two latent classes, we determine the vote combinations that lead to equal posterior probabilities. That is, we calculate the specific vote distribution $\boldsymbol{Y}^{(i)}$ such that $\tau_k = \tau_j$ for two classes $k, j \in \{1, 2, 3\}, k \neq j$, while there are no votes for the third class. This gives us the critical points lying on the axis connecting classes $k$ and $j$. For the middle point, i.e., the connection between all three classes, the equation $\tau_1 = \tau_2 = \tau_3$ is solved for the corresponding vote distribution. This results in four critical points. By connecting the points on the axes to the center, we obtain the new borders of the latent classes, which are now based on posterior probability estimates and not just on the empirical distribution of the votes for one instance. In other words, they are estimated by taking all data into account. The exact border points are described in Appendix A.

In Figure 1, for all instances that lie between the black and red borders, the latent class label does not agree with the majority vote. It is especially evident that the latent class *neutral* comprises a smaller fraction of vote distributions than it would have by majority voting (black line). More precisely, considering all cases with a majority vote for *neutral*, our model agrees for 83.3%, however, *entailment* is estimated for 6.9% of cases and *contradiction* for the remaining 9.8%, i.e., 16.7% of the majority vote *neutral* are assigned a different label by our model. This is however desirable, as many votes for one of the more informative classes (*entailment* or *contradiction*) strongly speak for exactly those classes, even if there is no majority. For example, having 40 votes for *contradiction*, 60 for *neutral*, and none for *entailment*, indicates that *entailment* is unlikely. Likewise, if *neutral* would be the "true" latent class, at least some votes for entailment are expected. Thus, in this setting, a latent *contradiction* is most probable. Analogous reasoning can be applied for instances with many votes for *entailment*, without *entailment* as the majority. Further, we argue that negative votes by the annotators can be regarded as a stronger signal for the instance actually being *contradiction* as fewer of them are required for our model to assign the label *contradiction*, compared to *entailment*. This becomes evident from Figure 1 as the red border between *neutral* and *contradiction* is much closer to the *neutral* corner compared to its counterpart between *neutral* and *entailment*.

To summarize, the model especially refines the class *neutral* and alleviates the issue that the majority class *neutral* does not only contain true neutral statements, but might also be conflated with examples where the annotators were indecisive or had conflicting interpretations (Nighojkar et al., 2023).

### 5.2 Stability Analysis

Having provided a visualization tool that allows valuable insights into the dataset, we are now interested in the *stability* of the modeling procedure. One common approach to assess the estimation uncertainty and stability of the resulting parameter estimates is to employ a resampling method, like bootstrapping (Efron, 1979). We therefore analyze the stability of the estimation procedure in relation to three aspects:

1. overall stability,
2. stability in the number of instances, $N$,
3. stability in the number of labels, $J$.

**Overall stability** In order to assess the uncertainty of the estimation procedure itself, we employ a classical bootstrap. That is, we sample from the data with replacement[2] and subsequently estimate the model parameters. Repeating this multiple times allows us to assess how the estimation would change if we had different datasets coming from the same distribution as the initial one.

[2] i.e., the same instance can be present multiple times, while other instances might not be included at all.

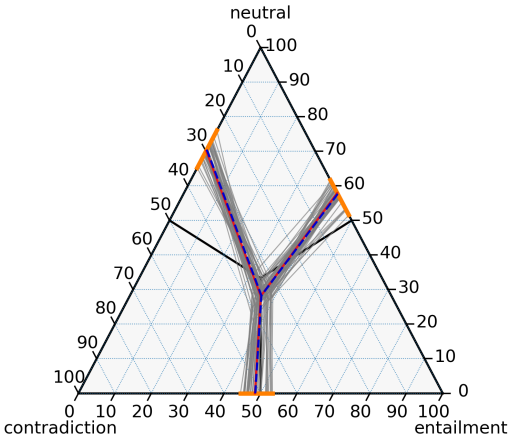



Figure 2: The ternary plot contains the decision borders between the three classes calculated based on $B = 50$ bootstrapped estimates as gray lines. The range of the gray lines is outlined in orange. The blue dashed line indicates the mean of the bootstrapped versions and the red line shows the original borders for comparison.

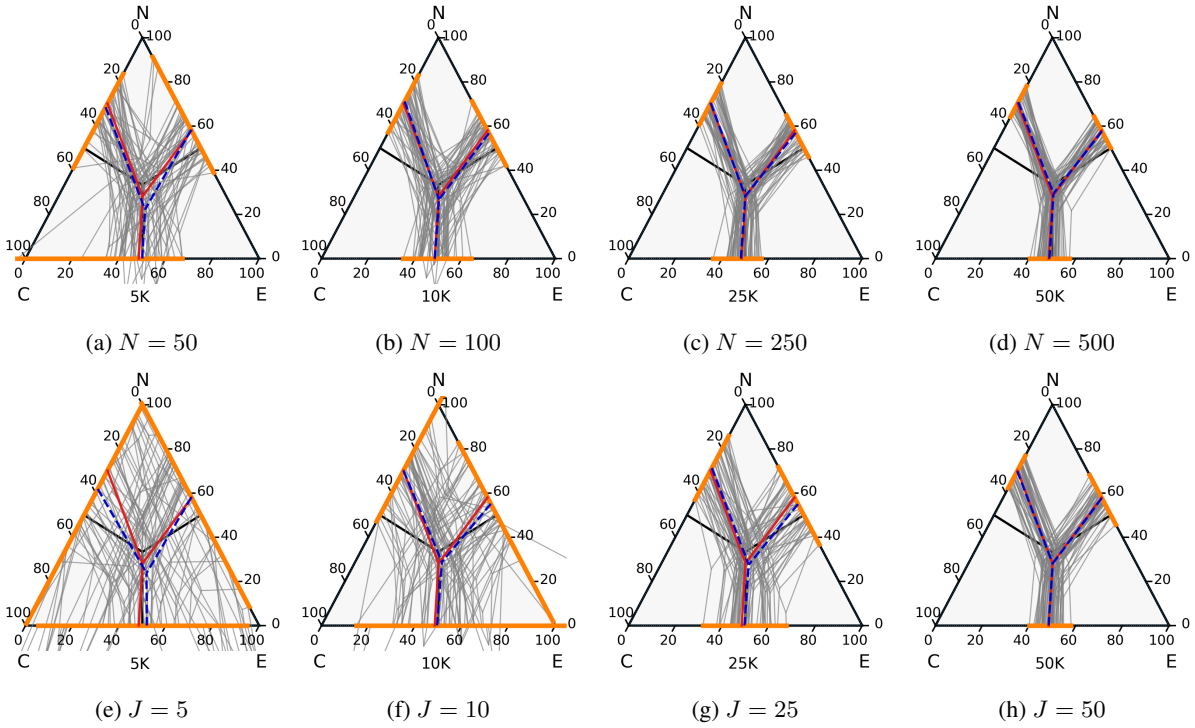
We run $B$ bootstrap iterations, producing bootstrapped versions of the parameter estimates $\boldsymbol{\pi}$ and $\boldsymbol{\Theta}$. Based on these values, the borders of the latent classes can be recalculated $B$ times. Figure 2 shows the estimated borders for $B = 50$ bootstrap replicates in gray alongside the borders computed based on the full dataset in red (cf. Fig. 1). This leads us to conclude that the estimation of the parameters and, therefore, the latent classes is stable for the full dataset. Due to the high number of instances in the dataset, this result is not surprising. However, the question arises whether stable estimation is also possible with a smaller dataset. Reducing $N$, the number of multiple annotated instances on the one hand, and reducing $J$, the number of annotations for the instances on the other hand, could lead to substantially reduced labeling effort. Hence, these aspects will be analyzed in the following.

**Stability of Number of Instances** In many real-world applications, the number of instances that can be annotated multiple times is often limited to a couple of hundred instances (as an example, the earlier multi-annotated NLI dataset from Pavlick and Kwiatkowski (2019) contained five annotations for less than 500 instances as available in ChaosNLI). Therefore, it is worthwhile to examine the stability of the estimation procedure and the resulting estimates for a smaller dataset in terms of sample size (less than 1.5k instances). Specifically, we are interested in the location of the decision borders regarding the latent classes and their stability for fewer instances.

Therefore, we employ a bootstrap again but this time randomly sample smaller datasets, i.e., $N < 1,514$ with replacement to artificially reduce the sample size. Figure 3 shows $B = 50$ bootstrapped borders of the latent classes for various numbers of samples $N$ with fixed $J = 100$. While the bootstrapped borders still show quite some variation for very small sample sizes (e.g., $N = 50$), the average of all bootstrapped borders already aligns quite well with the original borders. For a sample size of $N = 100$, the variation has already decreased noticeably, and for even larger samples, like $N = 500$, which is only one-third of the original sample size, almost no differences to the original results are visible. Hence, we conclude that reducing the sample size leads to reasonably good and stable estimation results if a certain minimum of instances is kept.

**Stability of Number of Labels** While this work focuses on the ChaosSNLI dataset with $J = 100$ annotations, the original SNLI development dataset only contains five labels per instance. In practice, annotating instances many times is costly and might seem inefficient. Hence, we are also interested in the stability of the estimation procedure in terms of the number of labels as well as the *minimal* number of labels needed per instance for stable parameter estimates.

Again, we draw bootstrap samples from the original dataset. This time, the sample size is kept constant at $N = 1000$ but the number of annotations per sample is reduced. Therefore, we randomly choose $J < 100$ annotations from the original ones. The resulting bootstrapped borders are shown in Figure 3. As expected, only using $J = 5$ annotations leads to large variations and unstable results. For $J = 25$ annotations, the procedure is already quite stable. For more than $J = 50$ annotations, the results show diminishing returns: they depict similar behavior to the original ones with the double amount of $J$, i.e., $J = 100$ (see Fig. 2). Therefore, we note that acquiring a smaller number of labels for each instance is possible, but a sufficient amount of annotations is needed for stable estimation. Particularly, the number of annotations seems to be more crucial for the stability of the results than the sample size. Additional results for simultaneously varying the amount of $N$ and $J$ that further support this finding can be found in Figure 4, Appendix A.




(a) $N = 50$ (b) $N = 100$ (c) $N = 250$ (d) $N = 500$

(e) $J = 5$ (f) $J = 10$ (g) $J = 25$ (h) $J = 50$

Figure 3: The ternary plots show the bootstrapped latent class borders as gray lines, the range of the gray lines in orange, the mean of the bootstrapped as blue dashed lines, and for comparison, the original borders as red lines for various sample sizes and annotations. In the top row $J$ is set to $J = 100$ and $N \in \{50, 100, 250, 500\}$. In the bottom row, we set $N = 1000$ and $J \in \{5, 10, 25, 50\}$. The total number of annotations, i.e., $N \cdot J$, is below each plot.

## 6 Discussion

Reliable and correct labels are crucial for classification models. While it is common practice to gather multiple annotations to ensure high-quality labels, these are often summarized into one single final label via a majority vote (Paun et al., 2018). However, this strategy leads to a major loss of information and uncertain ground truth labels in applications where a high degree of label variation is present. The statistical approach pursued in this work offers the possibility to condense information, given in multiple labels through the whole dataset, into a single ground truth label. To evaluate the results, we compared the borders between the classes, i.e., we examined the voting combinations where the ground truth label changes for an instance. By choosing the estimated latent ground truth instead of the majority vote, these borders shifted reasonably, from a semantic perspective.

Additionally, we showed that the parameters of the model and, hence, the borders can be estimated reliably based on the available instances and annotations. However, in many realistic applications, the data basis might be smaller in terms of both aspects. Hence, we also conducted a stability analysis for random subsets of the number of instances ($N$) and the number of votes per instance ($J$) of the dataset. The results show that stable estimation is already possible for a smaller dataset and that human labeling effort can be decreased, without loss of information. The quantity of accessible labels proves to be more important for ensuring a stable model performance than the sample size. We assume that this is because the annotations bear the majority of the inherent uncertainty. Therefore, acquiring multiple labels, particularly for uncertain instances, i.e., instances where label variation is expected, is advisable.

While the results and decision borders obtained via the proposed model in this work showcase the problem of label uncertainty, future directions of research could include the incorporation of this information into the ML pipeline or the development of a quantitative measure for label uncertainty. This could then lead to a detailed strategy for acquiring labels efficiently. Though these questions are highly relevant and should be tackled in the future, they are beyond the scope of the current work.

## 7 Conclusion

In conclusion, by analyzing ChaosSNLI we showcase the suitability of Bayesian mixture models to recover the true data-generating process of annotation tasks with access to multiple labels. Our work provides a framework to deal with multi-annotation settings in classification and is applicable regardless of the underlying task, i.e., NLI. Furthermore, our results suggest that in the annotation process, the focus should lie on increasing the number of labels per instance, instead of more instances in total, as this promotes capturing the labeling uncertainty.

## Limitations

Our proposed method analyzes uncertainty in labels for a three-way classification task. However, since the concept of *uncertainty* is by definition vague and fuzzy, it is important to determine which aspects of uncertainty *should be* or *can be* specified. In our work, we focus on modeling the annotation process. If other aspects of uncertainty are of relevance, our method might not be the most appropriate anymore. This points to the individuality of dealing with uncertainty and that no one-fits-all approach exists.

Further limitations might arise upon the application of the model to other datasets. 1) Multiple annotations per instance are needed. 2) Visual assessment of class memberships (c.f. Fig 1) or the stability of class borders (c.f. Fig 3) works reasonably well for up to three classes. Analyzing datasets with labels of higher dimensions is straightforward, as shown by Hechinger et al. (2024) for the classification of ambiguous images. However, assessing the stability of class borders needs to be done quantitatively, e.g., by computing confidence intervals of the bootstrapped borders. 3) In case annotator IDs are available, we recommend extending our approach in order to incorporate all available information. This could be done by determining the impact of individual annotators or a general annotator effect on the results, e.g., by discarding votes by certain annotators and re-estimating the model, see Hechinger et al. (2024).

Our work contributes to the understanding of NLI tasks and provides guidance for the early stage of data collection. Therefore, analyzing the impact on the full machine learning pipeline, i.e., improvements on the predictive power of classifiers is beyond the scope of this paper, but is open for future work.


## Acknowledgements

CG is supported by the DAAD program Konrad Zuse Schools of Excellence in Artificial Intelligence, sponsored by the Federal Ministry of Education and Research. KH is supported by the Helmholtz Association under the joint research school HIDSS-006 - Munich School for Data Science@Helmholtz, TUM&LMU. MA has been partially funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) as part of BERD@NFDI - grant number 460037581. BP is supported by European Research Council (ERC) grant agreement No. 101043235.


## References


Joris Baan, Nico Daheim, Evgenia Ilia, Dennis Ulmer, Haau-Sing Li, Raquel Fernández, Barbara Plank, Rico Sennrich, Chrysoula Zerva, and Wilker Aziz. 2023. Uncertainty in natural language generation: From theory to applications.

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

Arthur P Dempster, Nan M Laird, and Donald B Rubin. 1977. Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society: series B (methodological)*, 39(1):1–22.

B. Efron. 1979. Bootstrap methods: Another look at the jackknife. *The Annals of Statistics*, 7(1):1 – 26.

Cornelia Gruber, Patrick Oliver Schenk, Malte Schierholz, Frauke Kreuter, and Göran Kauermann. 2023. Sources of Uncertainty in Machine Learning – A Statisticians' View. ArXiv:2305.16703 [cs, stat].

Katharina Hechinger, Xiao Xiang Zhu, and Göran Kauermann. 2024. Categorising the world into local climate zones: towards quantifying labelling uncertainty for machine learning models. *Journal of the Royal Statistical Society Series C: Applied Statistics*, 73:143–161.

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

Eyke Hüllermeier and Willem Waegeman. 2021. Aleatoric and epistemic uncertainty in machine learning: an introduction to concepts and methods. *Machine Learning*, 110(3):457–506.

Nan-Jiang Jiang and Marie-Catherine de Marneffe. 2022. Investigating reasons for disagreement in natural language inference. *Transactions of the Association for Computational Linguistics*, 10:1357–1374.

Nan-Jiang Jiang, Chenhao Tan, and Marie-Catherine de Marneffe. 2023. Ecologically valid explanations for label variation in NLI. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10622–10633, Singapore. Association for Computational Linguistics.

Yixin Nie, Xiang Zhou, and Mohit Bansal. 2020. What Can We Learn from Collective Human Opinions on Natural Language Inference Data? ArXiv:2010.03532 [cs].

Animesh Nighojkar, Antonio Laverghetta Jr., and John Licato. 2023. No strong feelings one way or another: Re-operationalizing neutrality in natural language inference. In *Proceedings of the 17th Linguistic Annotation Workshop (LAW-XVII)*, pages 199–210, Toronto, Canada. Association for Computational Linguistics.

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

Ellie Pavlick and Tom Kwiatkowski. 2019. Inherent Disagreements in Human Textual Inferences. *Transactions of the Association for Computational Linguistics*, 7:677–694. Place: Cambridge, MA Publisher: MIT Press.

Barbara Plank. 2022. The 'Problem' of Human Label Variation: On Ground Truth in Data, Modeling and Evaluation. ArXiv:2211.02570 [cs].

Matthew Stephens. 2000. Dealing with label switching in mixture models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 62(4):795–809.

Alexandra N. Uma, Tommaso Fornaciari, Dirk Hovy, Silviu Paun, Barbara Plank, and Massimo Poesio. 2021. Learning from Disagreement: A Survey. *Journal of Artificial Intelligence Research*, 72:1385–1470.

Shujian Zhang, Chengyue Gong, and Eunsol Choi. 2021. Learning with Different Amounts of Annotation: From Zero to Many Labels. ArXiv:2109.04408 [cs].

Xinliang Frederick Zhang and Marie-Catherine de Marneffe. 2021. Identifying inherent disagreement in natural language inference. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4908–4915, Online. Association for Computational Linguistics.

# A Appendix

## Details on Model and Estimation

The EM algorithm is initialized with $\boldsymbol{\pi}_{(0)} = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$, and $\boldsymbol{\Theta}_{(0)}$ is drawn from a Dirichlet distribution where $\boldsymbol{\alpha}$ is set to be a vector with $K$ entries, where each value is $2 \cdot K$. In this case $\boldsymbol{\alpha} = (6, 6, 6)$.

The model estimated on the full dataset (i.e., $N = 1514, J = 100$), which is also depicted in Figure 1, results the following final parameter estimates:

$$\hat{\boldsymbol{\pi}} = (0.314, 0.448, 0.238)$$

$$\hat{\boldsymbol{\Theta}} = \begin{pmatrix} \hat{\boldsymbol{\theta}}_{entailment} \\ \hat{\boldsymbol{\theta}}_{neutral} \\ \hat{\boldsymbol{\theta}}_{contradiction} \end{pmatrix} = \begin{pmatrix} 0.73 & 0.24 & 0.03 \\ 0.14 & 0.79 & 0.07 \\ 0.03 & 0.31 & 0.66 \end{pmatrix}$$

In both parameters, the order of entries/columns is *entailment*, *neutral*, *contradiction*.

Based on the estimated parameters obtained via the procedure described in section 4 the decision borders are defined by connecting the points ([E, N, C]):

- center point: $[35.98, 28.15, 35.86]$
- EC axis: $[48.46, 0.0, 51.54]$
- EN axis: $[42.03, 57.97, 0.0]$
- NC axis: $[0.0, 70.13, 29.87]$

## Combined Stability Analysis

Figure 4 shows the estimation results and their bootstrapped stability for various sample sizes and numbers of annotations. Reducing $N$ and $J$ simultaneously leads to unstable results for very small datasets. However, this visualization supports the earlier finding that a sufficient number of annotations is more crucial than a large sample for stable and reliable estimation.

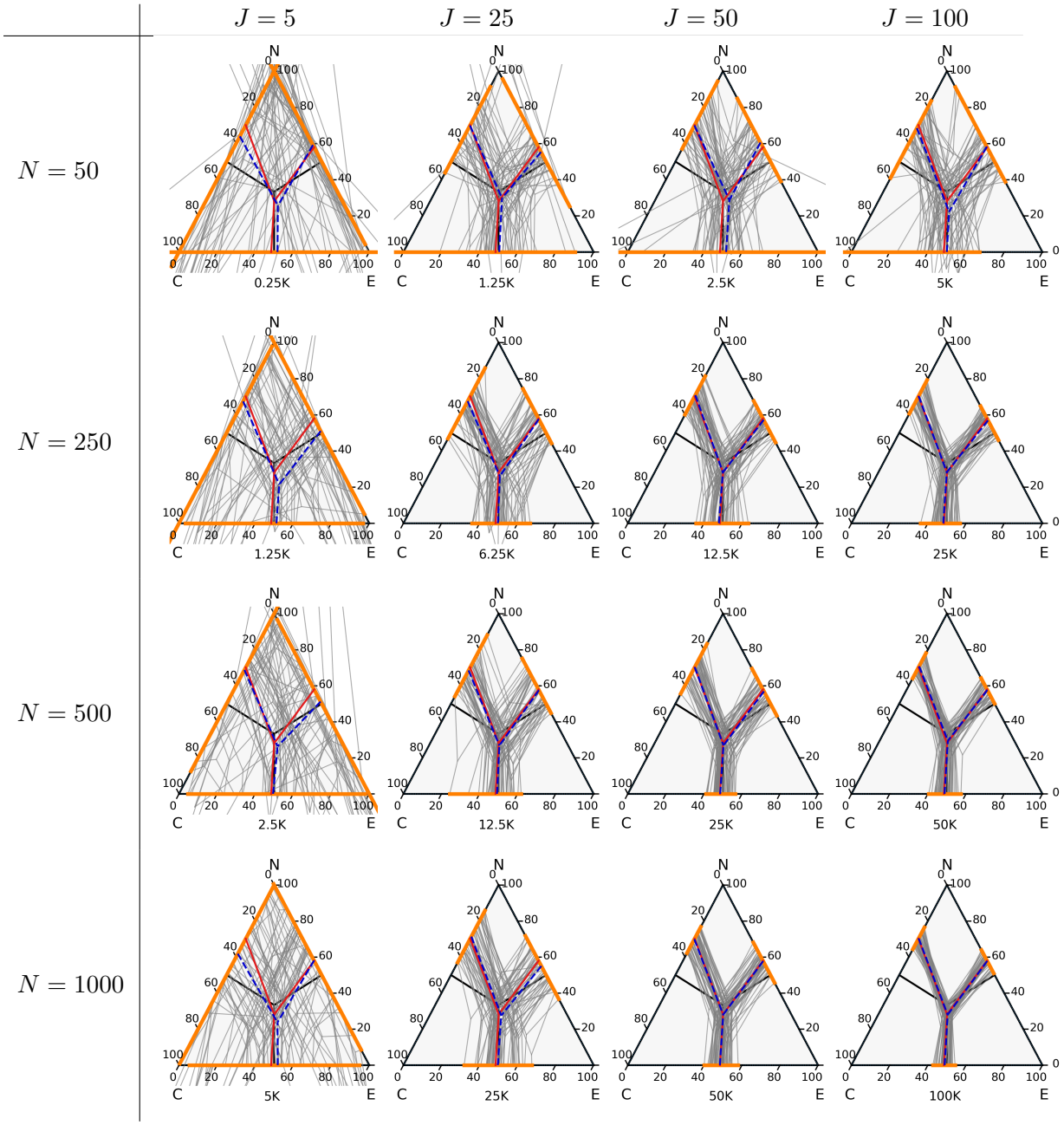



Figure 4: The Figure shows the bootstrapped latent class borders as gray lines, the range of the gray lines in orange, the mean of the bootstraps as blue dashed lines and the original borders as red lines for different values of $N$ and $J$. The total number of annotations, i.e., $N \cdot J$, is below each plot.