Natural Language Inference with Mixed Effects

William Gantt
University of Rochester

Benjamin Kane
University of Rochester

Aaron Steven White
University of Rochester

Abstract

There is growing evidence that the prevalence of disagreement in the raw annotations used to construct natural language inference datasets makes the common practice of aggregating those annotations to a single label problematic. We propose a generic method that allows one to skip the aggregation step and train on the raw annotations directly without subjecting the model to unwanted noise that can arise from annotator response biases. We demonstrate that this method, which generalizes the notion of a *mixed effects model* by incorporating *annotator random effects* into any existing neural model, improves performance over models that do not incorporate such effects.

1 Introduction

A common method for constructing natural language inference (NLI) datasets is (i) to generate text-hypothesis pairs using some method—commonly, crowd-sourced hypothesis elicitation given a text from some existing resource (Bowman et al., 2015; Williams et al., 2018) or automated text-hypothesis generation (Zhang et al., 2017); (ii) to collect crowd-sourced judgments about inference from the text to the hypothesis; and (iii) to aggregate the possibly multiple annotations provided for a single text-hypothesis pair into a single label. This final step follows common practice across annotation tasks in NLP; but for NLI in particular, there is growing evidence that it is problematic due to disagreement among annotators that is not captured by the probabilistic outputs of standard NLI models (Pavlick and Kwiatkowski, 2019).

One way to capture this disagreement would be to directly model the variability in the raw annotations. But this approach presents a challenge: it can be difficult to assess how much disagreement arises from disagreement about the interpretation of a text-hypothesis pair and how much is due to biases

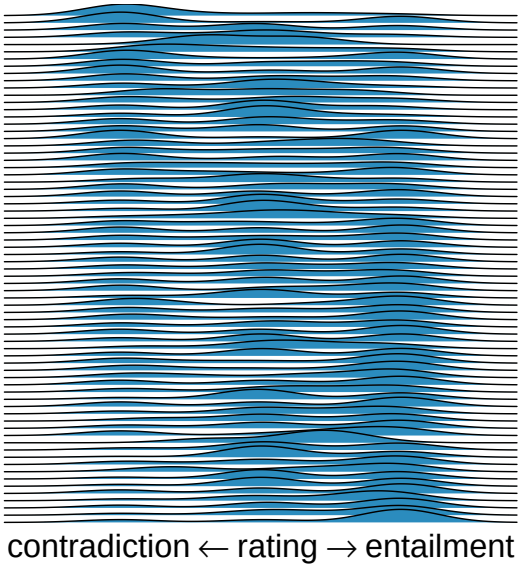

Figure 1: Distribution of [-50, 50] slider ratings (by annotator) for the same 20 NLI pairs in Pavlick and Kwiatkowski's dataset (batch 1, described in their §3).

that annotators bring to the task. Such biases can be extreme. For instance, Figure 1 plots the distribution by annotator of [-50, 50] ratings—with -50 clear contradiction and 50 clear entailment—for the same 20 NLI pairs in Pavlick and Kwiatkowski's dataset. Despite describing responses to the same items, the distributions are quite variable, suggesting variability in how annotators approach the task. This difference in approach may be relatively shallow—e.g. given some true label (or distribution thereon), annotators merely differ in their mapping of that value to the response scale—or they may be quite deep—e.g. annotators differ in how they interpret the relationship between texts and hypotheses.

We investigate both of these possibilities within a *mixed effects modeling* framework (Gelman and Hill, 2014). The core idea is to incorporate annotator-specific parameters into standard NLI models that either (i) merely modify the output of a standard classification/regression head or (ii) modify the parameters of the head itself. These

two options correspond to the mixed effects modeling concepts of *random intercepts* and *random slopes*, respectively. For the same reason that such *random effects* can be incorporated into effectively any generalized linear model in a modular way, our components can be be similarly incorporated into any NLI model. We describe how this can be done for a simple RoBERTa-based NLI model.

We find (i) that models containing only random intercepts outperform both standard models and models containing random slopes when annotators are known; and (ii) that when annotators are not known, performance drops precipitously for both random effects models. Together, these findings suggest that those building NLI datasets should provide annotator information and that those developing NLI systems should incorporate random effects into their models.

2 Extended Task Definition

In the standard supervised setting, NLI datasets are (graphs of) functions from text-hypothesis pairs $\langle T_i, H_i \rangle \in \Sigma^* \times \Sigma^*$ to inference labels $y_i \in \mathcal{Y}$—where $\mathcal{Y}$ is commonly {*contradicted*, *neutral*, *entailed*} or {*not-entailed*, *entailed*}, but may also be a finer-grained (e.g. five-point) ordinal scale (Zhang et al., 2017) or bounded continuous scale (Chen et al., 2020). The NLI task is to produce a single label from $\mathcal{Y}$ given a text-hypothesis pair.

We extend this setting by assuming that NLI datasets are (graphs of) functions from text-hypothesis pairs *and* annotator identifiers $a_i \in \mathcal{A}$ to inference labels and that the NLI task is to produce a single label given a text-hypothesis pair and an annotator identifier. A particular model need not make use of the annotator information during training and may similarly ignore it at evaluation time. Though many existing datasets do not provide annotator information, it is trivial for a dataset creator to add (even *post hoc*), and so this extension could feasibly be applied to any existing dataset.

3 Models

We assume some encoder that maps from $\langle T_i, H_i \rangle \in \Sigma^* \times \Sigma^*$ to $\langle \mathbf{x}_{T_i}, \mathbf{x}_{H_i} \rangle \in \mathbb{R}^M \times \mathbb{R}^N$ independently of annotator a_i, and we focus mainly on the mapping from $\mathbf{z}_i \equiv \langle \mathbf{x}_{T_i}, \mathbf{x}_{H_i} \rangle$ and a_i to y_i.

We consider two types of model: one containing only *annotator random intercepts* and another additionally containing *annotator random slopes*. The first assumes that differences among annotators are relatively shallow—e.g. given some true label for a pair (or distribution thereon), annotators have their own specific way of mapping that value to a response—and the second assumes that the differences among annotators are deeper—e.g. annotators differ in how they interpret the relation between texts and hypotheses. This distinction is independent of the labels $\mathcal{Y}$: regardless of whether the labels are discrete or continuous, random effects can be incorporated. In the language of generalized linear mixed models, the *link functions* are the only thing that changes. We consider two label types: three-way ordinal and bounded continuous.

Annotator random intercepts amount to annotator specific bias terms $\boldsymbol{\rho}_{a_i}$ on the raw predictions of a classification/regression head. Unlike standard *fixed* bias terms, however, what makes these terms random intercepts is that they are assumed to be distributed according to some prior distribution with unknown parameters. This assumption models the idea that annotators are sampled from some population, and it yields 'adaptive regularization' (McElreath, 2020), wherein the biases for annotators who provide few labels will be drawn more toward the central tendency of the prior.

Random intercepts for categorical outputs can take two forms, depending on whether the model enforces ordinality constraints—as linked logit models do (Agresti, 2014)—or not. Since most common categorical NLI models do not enforce ordinality constraints, we do not enforce them here, assuming that the model has some independently tunable function $h_{\boldsymbol{\theta}} : \mathbb{R}^M \times \mathbb{R}^N \rightarrow \mathbb{R}^{|\mathcal{Y}|}$ that produces potentials for each label and that:

$$f(y_i \mid \mathbf{z}_i, \boldsymbol{\theta}, \boldsymbol{\rho}_{a_i}) = \text{softmax}\left(h_{\boldsymbol{\theta}}(\mathbf{z}_i) + \boldsymbol{\rho}_{a_i}\right)$$

where $\boldsymbol{\rho}_{a_i} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$ with unknown $\boldsymbol{\Sigma}$.

Random intercepts for continuous outputs are effectively shifting terms on the single value predicted by some independently tunable function $h : \mathbb{R}^M \times \mathbb{R}^N \rightarrow \mathbb{R}$. If the continuous output is furthermore bounded, a squashing function g is necessary. In the bounded case, we assume that the variable—scaled to (0, 1)—is distributed Beta (following Sakaguchi and Van Durme, 2018) with mean μ_i and precision $\nu_i = \exp\left(\rho_{a_i 1} + \nu_0\right)$.

$$\mu_i = g\left(h_{\boldsymbol{\theta}}(\mathbf{z}_i) + \rho_{a_i 2}\right)$$

$$\alpha_i;\ \beta_i = \mu_i \nu_i;\ (1 - \mu_i)\nu_i$$

$$f(y_i \mid \mathbf{z}_i, \boldsymbol{\theta}, \boldsymbol{\rho}_{a_i}; \nu_0) = \text{Beta}(y_i \mid \alpha_i, \beta_i)$$

where $\boldsymbol{\rho}_{a_i} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$ with unknown $\boldsymbol{\Sigma}$. This implies that $\nu_i \sim \log \mathcal{N}(\nu_0, \sigma_{11}^2)$ with unknown ν_0.

MegaVeridicality
▶ *Someone knew that something happened.*
That thing happened.
▶ *Someone thought that something happened.*
That thing happened.
MegaNegRaising
▶ *Someone didn't think that something happened.*
That person thought that thing didn't happen.
▶ *Someone didn't know that something happened.*
That person knew that thing didn't happen.

Table 1: NLI sentence pairs from MegaVeridicality and MegNegRaising. ▶ indicates the line is a text, and the following line is its corresponding hypothesis. Hypotheses in green indicate that the context entails the hypothesis; those in red indicate that it does not.

The precision parameter ν_i controls the shape of the Beta: with small ν_i, a_i tends to give responses near 0 and 1 (whichever is closer to μ_i); with large ν_i, a_i tends to give responses near μ_i.

Annotator random slopes amount to annotator-specific classification/regression heads $h_{\boldsymbol{\phi}_i}$. We swap these heads into the above equations in place of $h_{\boldsymbol{\theta}}$. As for the random intercept parameters, we assume that the annotator-specific parameters $\boldsymbol{\phi}_i$, which we refer to as the annotator random slopes, are distributed $\boldsymbol{\phi}_i \sim \mathcal{N}(\boldsymbol{\theta}, \boldsymbol{\Sigma})$ with unknown $\boldsymbol{\theta}, \boldsymbol{\Sigma}$. One way to think about this model is that $h_{\boldsymbol{\theta}}$ produces prototypical interpretation around which annotators' actual interpretations are distributed.

4 Experiments

We compare models both with and without random effects when fit to NLI datasets conforming to the extended setting described in §2. The model without random intercepts (the *fixed model*) simply ignores annotator information—effectively locking $\boldsymbol{\rho}_{a_i}$ to $\mathbf{0}$ for all annotators a_i.

Encoder All models use pretrained RoBERTa (Liu et al., 2019) as their encoder. We use the basic LM pretrained versions (no NLI fine-tuning).

Data To our knowledge, the only NLI datasets that both provide annotator identifiers and are large enough to train an NLI system are MegaVeridicality (MV; White and Rawlins, 2018; White et al., 2018), which contains three-way categorical annotations aimed at assessing whether different predicates give rise to veridicality inferences in different syntactic structures, and MegaNegRaising (MN; An and White, 2020), which contains bounded continuous [0, 1] annotations aimed at assessing whether different predicates give rise to neg(ation)-raising inferences in different syntactic structures. Table 1 shows example pairs from each dataset. Both datasets contain 10 annotations per text-hypothesis pair from 10 different annotators. MV contains 3,938 pairs (39,380 annotations), and MN contains 7,936 pairs (79,360 annotations). In both datasets, each pair is constructed to include a particular main clause predicate and a particular syntactic structure. To test each model's robustness to lexical and structure variability, we use this information to construct folds of the cross-validation (see **Evaluation**).

Classification/Regression Heads We consider heads with a single affine layer as well as one hidden affine layer followed by a rectifier. In the latter case, the hidden layer size is set to half the size of the input—i.e. with encoder output size K (768 for RoBERTa), the hidden layer has size $\frac{K}{2}$.

Loss We use the negative log-likelihood of the observed values under the model as the loss.

Evaluation We evaluate all of our models using 5-fold cross-validation. We consider four partitioning methods: (i) RANDOM: completely random partitioning; (ii) PREDICATE: partitioning by the main clause predicate found in the text (a particular main clause predicate occurs in one and only one partition); (iii) STRUCTURE: partitioning by the syntactic structure found in the text (a particular structure occurs in one and only one partition); and (iv) ANNOTATOR: a particular annotator occurs in one and only one partition. For the first three methods, we ensure that each annotator occurs in every partition, so that random intercepts and random slopes for that annotator can be estimated. For the ANNOTATOR method, where we do not have an estimate for the random effects of annotators in the held-out data, we use the mean of the prior.

We report mean accuracy on held-out folds for the categorical data (MV); and following Chen et al. (2020), we report mean rank correlation on held-out folds for the bounded continuous data (MN). To make these metrics comparable, we report them relative to the performance of both a baseline model and the best possible fixed model.

$$\text{score}_{\text{mod}} = \frac{\text{raw-score}_{\text{mod}} - \text{raw-score}_{\text{base}}}{\text{raw-score}_{\text{best}} - \text{raw-score}_{\text{base}}}$$

For the categorical data, the baseline model predicts the majority class across all pairs, and the best possible fixed model predicts the majority class across annotators for each pair. Similarly, for the bounded continuous data, the baseline model predicts the mean response across all pairs, and the best possible fixed model predicts the mean response across annotators for each pair.

Model	RANDOM		PREDICATE		STRUCTURE		ANNOTATOR	
	Acc	*Corr*	*Acc*	*Corr*	*Acc*	*Corr*	*Acc*	*Corr*
Fixed	0.93	0.32	0.92	0.23	0.80	0.09	**0.91**	0.27
Random Intercepts	**1.23**	**1.29**	**1.26**	**1.92**	**1.07**	**1.71**	-0.89	**0.41**
Random Slopes	0.89	1.11	1.00	1.61	0.62	1.42	-0.86	0.07

Table 2: Mean of the rescaled accuracy (categorical data) and rank correlation (bound continuous data) across cross-validation folds for each partitioning method ($\text{score}_{\text{mod}}$ from §4). Bolded values are best in column.

These relative scores are 0 when the model does not outperform the baseline and 1 when the model performs as well as the best possible fixed model. It is possible for a random effects model to obtain a score of greater than 1 by leveraging annotator information or less than 0 if it overfits the data.

5 Results

Table 2 shows the results of our experiments. The random intercepts models reliably outperform the fixed models in all cross-validation settings except ANNOTATOR in Bonferroni-corrected Wilcoxon rank-sum tests ($\alpha = 0.05$). Indeed, they tend to reliably outperform even the best possible fixed model, having rescaled scores above 1. The results for the random slopes models are more mixed: sometimes they outperform the fixed model, while other times they do not; and they similarly show a drop in performance in the ANNOTATOR setting.

Consistent with Pavlick and Kwiatkowski's findings, these results suggest that variability in annotators' responding behavior is substantial; otherwise, it would not be possible for the random effects models to outperform the best possible fixed model, and we would not expect such precipitous drops in performance when annotator information is removed. But this variability is likely relatively shallow: if these differences were due to deeper differences in annotators' interpretation of the pair, we expect that the random slopes model would have at least done as well as the random intercepts model (since the random slopes model subsumes the random intercepts model); but there is clear evidence of overfitting on the part of the random slopes model. Of course, it remains a live possibility that the encoder we used does not extract features that are linearly related to the relevant interpretive variability, and so further investigation of random slopes models with different encoders may be warranted.

Contrasting the results on ordinal and bounded continuous data, the fixed model tends to perform better on ordinal data, while a similar trend is not seen for the random effects models. Indeed, the random intercepts model performs substantially better on the bounded continuous data under the PREDICATE and STRUCTURE settings. These results could be due to the link function we used for the bounded continuous data: the fixed model consistently learned small values for the precision parameter ν_0, resulting in sparse (bimodal) beta distributions. But the fact that the random intercepts model reliably outperforms the best possible fixed model implies that any tweaks to the link function would not bring the fixed model up to the level of the random intercepts model.

6 Related Work

The models developed here are closely related to models from Item Response Theory (IRT). IRT has been used to assess annotator quality (Hovy et al., 2013, 2014; Rehbein and Ruppenhofer, 2017; Paun et al., 2018a,b; Zhang et al., 2019; Felt et al., 2018) and various properties of an item (Passonneau and Carpenter, 2014; Sakaguchi and Van Durme, 2018; Card and Smith, 2018), including difficulty (Lalor et al., 2016, 2018, 2019). Other non-IRT-based work attempts to measure the relationship between annotator disagreement and item difficulty (Plank et al., 2014; Kalouli et al., 2019).

7 Conclusion

We find (i) that models containing only random intercepts outperform standard models when annotators are known, and (ii) that models that further contain random slopes do not yield any additional benefit. These results indicate that, though differences among NLI annotators' response behavior are important to model, these differences may not be particularly deep, limited to the ways in which annotators use the response scale, but not relating to deeper interpretive differences. We also find that removing annotator information decreases performance substantially, suggesting that NLI dataset developers should provide annotator information and NLI system developers should incorporate random effects into their models.

References

Alan Agresti. 2014. *Categorical Data Analysis*. John Wiley & Sons.

Hannah An and Aaron White. 2020. The lexical and grammatical sources of neg-raising inferences. *Proceedings of the Society for Computation in Linguistics*, 3(1):220–233.

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.

Dallas Card and Noah A. Smith. 2018. The importance of calibration for estimating proportions from annotations. 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 1636–1646, New Orleans, Louisiana. Association for Computational Linguistics.

Tongfei Chen, Zhengping Jiang, Adam Poliak, Keisuke Sakaguchi, and Benjamin Van Durme. 2020. Uncertain natural language inference. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8772–8779, Online. Association for Computational Linguistics.

Paul Felt, Eric Ringger, Kevin Seppi, and Jordan Boyd-Graber. 2018. Learning from measurements in crowdsourcing models: Inferring ground truth from diverse annotation types. In *International Conference on Computational Linguistics*.

Andrew Gelman and Jennifer Hill. 2014. *Data Analysis Using Regression and Multilevel/Hierarchical Models*. Cambridge University Press, New York City.

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

Dirk Hovy, Barbara Plank, and Anders Søgaard. 2014. Experiments with crowdsourced re-annotation of a POS tagging data set. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 377–382, Baltimore, Maryland. Association for Computational Linguistics.

Aikaterini-Lida Kalouli, Annebeth Buis, Livy Real, Martha Palmer, and Valeria de Paiva. 2019. Explaining simple natural language inference. In *Proceedings of the 13th Linguistic Annotation Workshop*, pages 132–143, Florence, Italy. Association for Computational Linguistics.

John P. Lalor, Hao Wu, Tsendsuren Munkhdalai, and Hong Yu. 2018. Understanding deep learning performance through an examination of test set difficulty: A psychometric case study. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4711–4716, Brussels, Belgium. Association for Computational Linguistics.

John P. Lalor, Hao Wu, and Hong Yu. 2016. Building an evaluation scale using item response theory. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 648–657, Austin, Texas. Association for Computational Linguistics.

John P. Lalor, Hao Wu, and Hong Yu. 2019. Learning latent parameters without human response patterns: Item response theory with artificial 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 4249–4259, Hong Kong, China. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv:1907.11692 [cs]*. ArXiv: 1907.11692.

Richard McElreath. 2020. *Statistical Rethinking: A Bayesian course with examples in R and Stan*. CRC Press.

Rebecca J. Passonneau and Bob Carpenter. 2014. The benefits of a model of annotation. *Transactions of the Association for Computational Linguistics*, 2:311–326.

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

Silviu Paun, Jon Chamberlain, Udo Kruschwitz, Juntao Yu, and Massimo Poesio. 2018b. A probabilistic annotation model for crowdsourcing coreference. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1926–1937, Brussels, Belgium. Association for Computational Linguistics.

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

Barbara Plank, Dirk Hovy, and Anders Søgaard. 2014. Linguistically debatable or just plain wrong? In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short*

Papers), pages 507–511, Baltimore, Maryland. Association for Computational Linguistics.

Ines Rehbein and Josef Ruppenhofer. 2017. Detecting annotation noise in automatically labelled data. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1160–1170, Vancouver, Canada. Association for Computational Linguistics.

Keisuke Sakaguchi and Benjamin Van Durme. 2018. Efficient online scalar annotation with bounded support. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 208–218, Melbourne, Australia. Association for Computational Linguistics.

Aaron Steven White and Kyle Rawlins. 2018. The role of veridicality and factivity in clause selection. In *Proceedings of the 48th Annual Meeting of the North East Linguistic Society*, pages 221–234, Amherst, MA. GLSA Publications.

Aaron Steven White, Rachel Rudinger, Kyle Rawlins, and Benjamin Van Durme. 2018. Lexicosyntactic inference in neural models. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4717–4724, Brussels, Belgium. Association for Computational Linguistics.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. 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 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Sheng Zhang, Rachel Rudinger, Kevin Duh, and Benjamin Van Durme. 2017. Ordinal common-sense inference. *Transactions of the Association for Computational Linguistics*, 5:379–395.

Yi Zhang, Zachary Ives, and Dan Roth. 2019. Evidence-based trustworthiness. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 413–423, Florence, Italy. Association for Computational Linguistics.