Interface Design for Crowdsourcing Hierarchical Multi-Label Text Annotations

Rickard Stureborg
Duke University
Durham, NC, USA
rickard.stureborg@duke.edu

Bhuwan Dhingra
Duke University
Durham, NC, USA
bdhingra@cs.duke.edu

Jun Yang
Duke University
Durham, NC, USA
junyang@cs.duke.edu

ABSTRACT

Human data labeling is an important and expensive task at the heart of supervised learning systems. Hierarchies help humans understand and organize concepts. We ask whether and how concept hierarchies can inform the design of annotation interfaces to improve labeling quality and efficiency. We study this question through annotation of vaccine misinformation, where the labeling task is difficult and highly subjective. We investigate 6 user interface designs for crowdsourcing hierarchical labels by collecting over 18,000 individual annotations. Under a fixed budget, integrating hierarchies into the design improves crowdsource workers' F1 scores. We attribute this to (1) Grouping similar concepts, improving F1 scores by +0.16 over random groupings, (2) Strong relative performance on high-difficulty examples (relative F1 score difference of +0.40), and (3) Filtering out obvious negatives, increasing precision by +0.07. Ultimately, labeling schemes integrating the hierarchy outperform those that do not — achieving mean F1 of 0.70.

CCS CONCEPTS

• **Human-centered computing** → *HCI design and evaluation methods.*

KEYWORDS

crowdsourcing, text annotation, user experience design

ACM Reference Format:
Rickard Stureborg, Bhuwan Dhingra, and Jun Yang. 2023. Interface Design for Crowdsourcing Hierarchical Multi-Label Text Annotations. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23), April 23–28, 2023, Hamburg, Germany.* ACM, New York, NY, USA, 17 pages. https://doi.org/10.1145/3544548.3581431

1 INTRODUCTION

To both build and evaluate machine learning systems, researchers often rely on human-labeled datasets [13, 41, 54]. Gathering this labeled data efficiently and at high quality is a well-studied problem when labels are binary [25, 34, 55] or a flat list of choices [14, 33, 47], but labels can often be grouped into other structures as well, such as species in a taxonomy [50].

CHI '23, April 23–28, 2023, Hamburg, Germany

ACM ISBN 978-1-4503-9421-5/23/04…$15.00
https://doi.org/10.1145/3544548.3581431

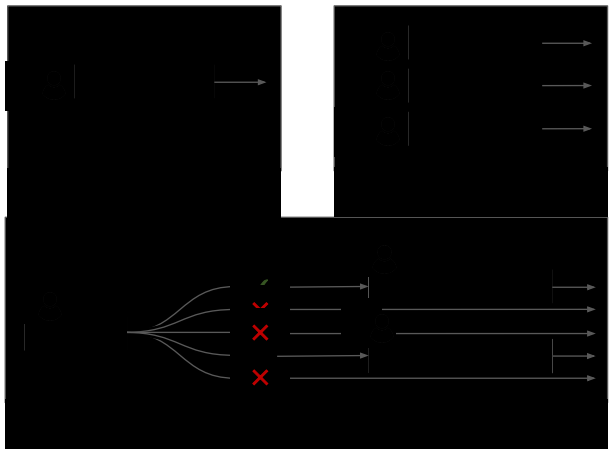

Figure 1: Pass-logic options when annotating a single passage. Each orange box represents a single question asked to workers (denoted A, B, ...) and "groups" refers to partitioning the labels into smaller sets of labels. Workers perform best on mean F1 score when using hierarchical multi-pass schemes.

Concept hierarchies (or taxonomies; ontologies) are used in many applications ([10, 17, 18]) to describe concepts at a flexible granularity, and generally serve to help organize and structure both language and thought around a topic. In certain situations, they may become a target for data labeling itself, where multiple hierarchically-structured class labels can be chosen for a given instance, a setting known as hierarchical multi-label annotation [7, 56]. Given their usefulness in organizing thought, one might expect that leveraging the hierarchy during annotation may yield higher quality and efficiency. However, there are many design choices to consider: Does interface complexity increase cognitive load ([19, 40])? Will false negatives in an upper level of the hierarchy end up amplifying errors into annotations on a lower level [7]? Will presenting one part of the hierarchy while hiding the rest create a lack of context that leads to misinterpreted label definitions?

In this paper, we study how to incorporate the concept hierarchy into labeling schemes for crowdsource data annotation platforms such as Amazon Mechanical Turk (AMT). We focus on a difficult annotation task, assigning vaccine concerns ([46]) to text passages taken from anti-vaccination websites. The hierarchy of vaccine concerns includes labels such as *"Health risks"* and *"Issues with research"*. Small, purpose-built taxonomies are common in the domain of misinformation research [1, 10, 24, 42]. In the setting of difficult annotation tasks, we show that labeling schemes incorpor-

ating hierarchies can help annotators perform better against ground-truth labels.

We investigate two separate design choices when annotating a single passage: (1) how to *format* the hierarchical labels when shown on the interface, and (2) the *pass-logic* that decides how to coordinate multiple workers towards labeling that passage. We compare two formats for presenting the hierarchy to annotators (see Figure 3):

- *multi-label*, which simply presents labels as a flat multiple choice list of options
- *hierarchical multi-label*, which presents the entire hierarchy directly to the worker who then marks all relevant labels

For pass-logic, we look at three options (see Figure 1):

- *single-pass*, where all labels are presented to a single worker, who annotates the passage on their own
- *multi-pass*, which combines multiple workers' annotations for a single passage by partitioning the labels into groups (each worker focuses on a small subset of labels at a time).
- *hierarchical multi-pass*, in which a preliminary stage of annotation determines if child-labels need to be annotated.

We compare all valid combinations of these formats and pass-logic options under a fixed-budget setting, which provides practical insights for research and engineering teams interested in data collection of hierarchical multi-label tasks. For multi-pass logic, we consider both randomly partitioning labels into smaller subsets or utilizing the groupings given to us by the hierarchy. Our results point to a few statistically significant factors:

(1) Grouping similar concepts together: When partitioning labels using the hierarchy as opposed to a random partition, we see significantly better performance for the groupings informed by the hierarchy (F1 score of 0.50 grouped vs 0.34 random)
(2) Relative performance boost on difficult examples: Explicit access to the hierarchy increases workers performance on more difficult questions (as much as a +0.40 in F1 as compared to multi-label).
(3) Boosting true positive frequencies: By filtering out irrelevant passages from stage 2 annotation, more of the examples shown to workers are therefore true positives, which we show is associated with better precision without a detriment to recall. The performance boost from this alone moves the F1 score from 0.50 to 0.57.

Our results lead us to believe that difficult, high-subjectivity labeling tasks warrant new recommendations separate from crowdsource design guidelines in previous work ([7, 21]). We recommend considering incorporating hierarchies into the labeling process, and show a few options for how to do so. This is especially true if optimizing for individual worker performance, while choice of labeling scheme plays less of a role if using aggregation methods across several copies of annotations.

2 RELATED WORKS

The reliance of supervised ML algorithms on labeled data has led to a great wealth of knowledge regarding efficient data labeling at large scale. Huge datasets have been constructed requiring immense human labeling time across many media. Among them are image and video datasets generally containing thousands of classes such as ImageNet (14M images) [13] and OpenImages (9M images) [30], but even with fewer classes, such as CelebFaces labeling 40 facial attributes (200k images) [34]. Also audio datasets, typically with hundreds of classes, for instance AudioSet (2M clips) [18], Free Music Archive (100k clips) [12], and OpenMIC-2018 (20k clips) [21] Lots of work is focused on allowing this scale of data collection while maintaining high quality [8, 14, 29, 51] or protecting crowdsource workers [4, 23].

Often, this labeling is done on tasks with low ambiguity or subjectivity, and minimal required training – which makes them suitable for large scale collection. For example, in ImageNet [13], labels are the names of well-known objects such as "ambulance", "folding chair" or "snail." Even in more difficult audio-annotation tasks such as labeling noise categories in a busy city ([7]), the labels ("jackhammer", "car horn") have strong, objective definitions.

Given the clarity on such label definitions, previous studies on user interface design for crowdsource annotation have recommended increasing annotation throughput, or the rate at which labels are collected from the annotation platform [4, 16, 37]. Throughput can be very quick for some tasks (minutes for hundreds to thousands of annotations), while other tasks may be much slower. Prior work found that single-pass methods have up to 9 times higher throughput if annotations are required to be fully labeled (assigning a value for every label) rather than sparse [7]. However, work in psychology has long known that there is a tradeoff between speed and accuracy for any information processing task a human performs [53]. Other HCI work also studies this tradeoff [35, 57]. This suggests optimizing for throughput could be harmful to annotation quality, particularly if the task is difficult.

The cognitive load theory [48] suggests that tasks with high cognitive load (the amount of mental effort) can induce errors and mistakes at higher frequency than tasks with lower cognitive load. Work on user interfaces which require some level of accuracy often tries to minimize unnecessary cognitive load [19, 40, 51]. Similarly, work in crowdsourcing recommends to chunk difficult tasks into smaller units of work [28]. Some work has shown that crowdsource platforms have great potential for rapidly collecting measurements in user studies [27]. Other work examines how long annotators remain on tasks, and characterizes differences between those that annotate few examples versus those that annotate many [15].

Recent efforts have also moved towards datasets for high-impact social issues such as: misinformation [10, 46], which attempts to classify common concerns regarding issues such as climate change or vaccines; fact-checking [49], which labels whether claims are verified by trusted sources; and claim review [2, 3], which determines if claims are worth fact-checking. Such labels inherently lend themselves to be a more difficult annotation task, given the subjective label definitions and necessary processing to parse written rationales or arguments in text.

In data labeling, it is common to collect multiple copies of annotations and aggregate them using a majority vote [5, 54]. Some work studies how to perform aggregation more effectively [52]. This is said to reduce the impact of low-quality annotations during collection. Some old work in aggregation methods such as EM uses

weightings from estimates of worker skill [11], while other work incorporates question difficulty through parametric approaches [26] or non-parametric approaches [44].

However, recent trends in NLP have began questioning aggregation, arguing that subjective labels should not be aggregated if multiple opinions are valid. Rather, this line of work ([38, 58]) suggests predicting the distribution of human opinions, rather than the majority vote. One implication that follows is that individual annotator performance becomes more important, since one cannot aggregate away labeling error using a simple majority vote.

Labels are not always in the form of lists. There has been a large amount of work on labeling hierarchical multi-label annotations [7, 21, 45, 56], where the task is to select any relevant option from labels in a hierarchical structure. While most work employs a small group of experts to build the concept hierarchies before it gets labeled by workers, some research attempts to build these hierarchies through crowdsourcing methods [6, 9].

In considering the performance of crowdsource workers, a lot of effort has been spent to introduce gamification of the labeling task [22, 31, 36, 39, 43], but we note that this requires significant overhead efforts to build the games, which may not be feasible when data collection is time-sensitive.

3 APPROACH

3.1 Data collection

We study interface designs when labeling against a taxonomy of vaccine concerns developed to promote high agreement among crowdsource workers [46]. The taxonomy is a hierarchy of labels with 5 top-level concerns such as *Untrustworthy actors* and *Health risks*, and 19 child labels such as *Untrustworthy actors* → *Profit motives*. See Appendix A for the full version of the taxonomy. The annotation task consists of annotating passages from known anti-vaccination blogs and websites, pre-filtered to ensure the articles are on the topic of vaccination, against multiple labels from both levels in the taxonomy. Articles are converted into paragraphs using existing markers in the HTML code to closely resemble the paragraphs rendered to readers. An example is shown in Figure 2.

> The Minister of Fear (the CDC) was working overtime peddling doom and gloom, knowing that frightened people do not make rational decisions — nothing sells vaccines like panic.

Figure 2: Example passage from an anti-vaccination blog. Here, the correct labels from the taxonomy include *Untrustworthy actors* → *Profit motives* since the mention of "selling" implies money is a corrupting motive, as well as *Lack of benefits* → *Insufficient risk* since "peddling doom" implies that the dangers of the disease are being exaggerated.

These blog articles are often written with vague mentions of these recurring themes of concerns, and paragraphs are given to annotators without context regarding who wrote it or what paragraph came before. There is therefore lots of ambiguity in the input text which must be dealt with by annotators. The authors had some disagreement initially in 45% of passages, an indication of the level of subjectivity existing in the task. This is not surprising, given the labeling task primarily revolves around a concept ripe with subjectivity: concerns. Passages may simply raise different concerns for different readers. Unlike the annotation of object in images, for example, there are very few passages where the correct labels are immediately obvious. That being said, such passages do occur—particularly when there is high overlap between the vocabulary used to define labels and the vocabulary in the passage.

3.2 Annotator training

To maximize the chance of high-quality annotations, we look into a few methods to train annotators and ensure quality. These methods are implemented through the exact same process for all labeling schemes to ensure fairness. We collect all our annotations on Amazon Mechanical Turk (AMT).

3.2.1 Definitions. We provide written definitions for all labels and set up a micro-task as the very first step to have workers interact with the definitions directly. The very first screen the annotators will see is a list of all the labels they are expected to select from. Under each label is a written definition. The task we ask workers to complete is to mark any definition which they feel is unclear. This hopes to prompt fully reading and internalizing the definitions, as well as collects data for further improving the training process.[1] (See Appendix E for a screenshot of this step)

3.2.2 Tutorial. Next, workers walk through 10 examples, where they annotate passages just like they would in real annotation. However, for these 10 examples, they are given corrections after each submission. The corrections show which labels they got wrong and which they got correct. For incorrectly marked labels, there is a written explanation for why the label should have been applied (or not). Tutorial explanations are written ahead of time, and appropriate tutorial examples are given according to which labels are presented to the worker. We ensure that there is always a consistent ratio of different types of examples in each tutorial. For example, there are always two examples where none of the labels should be selected, one where the passage is clearly anti-vaccination but no specific argument is made (e.g., "vaccines are bad") and one where the passage is off-topic.

3.2.3 Entrance exam. After finishing the tutorial but before being allowed to annotate real data, we have workers complete a three-question entrance exam. To workers, this looks like regular annotation. Two of these passages are clearly off-topic, and a third passage clearly mentions one of the concerns being labeled. If workers fail any one of these three questions they are banned from labeling.

3.2.4 Quality checks. While the annotations are being collected, we randomly include attention checks (with 5% probability) such as "Help us catch cheaters. Choose the first option and hit submit to show you are paying attention." If workers fail such attention checks, we throw out all the annotations they gave us since the last passed attention check, and ban them from further annotations.

[1] For this paper we do not alter the training process in order to control for this step across all labeling schemes

3.3 Ground-truth labels

To evaluate the different labeling schemes, we collect "expert" annotations from three authors of the paper. The sample size for evaluation spans 4,800 passage-label pairs (200 passages taken from 200 articles). First, the three authors annotate the passages separately, followed by a discussion phase in which they try to come to an agreement about diverging labels. We refer to these labels as the *ground truth*, and separate them into four categories: (1) labels which were agreed on immediately during individual, non-communicative annotation; (2) labels which were agreed on after re-annotating them individually without communicating, but asking for a written rationale for the given label; (3) labels which were agreed on after collaborative discussion; and (4) labels which never reached unanimous agreement, but rather a majority vote was taken. These categories can be seen as a proxy for difficulty, requiring increasing amounts of nuanced examination of the target passage. Further details on the construction of these sets can be found in Appendix M, and an analysis of the effect of difficulty in §4.3.2.

3.4 Labeling schemes

In this section, we discuss the definitions of each interface design through the two formats we consider (*multi-label* and *hierarchical multi-label*) but also a third option which we do not include in experiments due to prohibitive costs (*binary-label*). We then explore the three pass-logic options (*single-pass*, *multi-pass*, and *hierarchical multi-pass*) and show our design approach for combining these options.

3.4.1 Formats. Labels can be presented on an interface in many different formats (see Figure 3). Here, formats refers to how to organize the set of labels in the user interface.

- *Binary-label* format shows the label to annotators using a single yes/no question. A worker will focus on a single label across their time annotating, minimizing cognitive load. [2]
- *Multi-label*, which simply presents labels as a flat multiple choice list of options. The workers can select any/all/none of the labels. Depending on the pass-logic used, this list may be longer or shorter, but will generally only contain labels from the same level of the hierarchy.
- *Hierarchical multi-label*, which presents a hierarchy directly such that choices in the top-level of the hierarchy prompt further choices in the next level. This option can be accomplished in two ways. In one version (v1), the hierarchy is given as checkboxes with child-level checkboxes that become enabled only if the parent category is selected. In the other (v2), the hierarchy is asked in a two-stage question. First there is a binary choice regarding the parent category. If the answer is yes, then a flat list of checkboxes for the child labels is presented to the worker.

[2]This approach has shown useful for high-quality data annotation for images [32], but has been less successful in video and audio [7, 51] due to its high cost when there is a temporal element in the annotation.

Figure 3: User interface designs for presenting labels to the user. In single-pass schemes, using majority vote and the hierarchical multi-label v1 performs the best with a mean F1 score of 0.70.

3.4.2 Pass logic. Pass logic determines how many workers are brought together to work on the annotation of a single passage, and how to coordinate their efforts. Some decisions regarding pass logic will inform the look of the interface shown to users, while others will only affect which passages are shown to any given worker. We examine three options for pass logic (see Figure 1):

- *Single-pass*, where one worker is asked to annotate the passage entirely on their own, and therefore must be presented all the labels at once. This option can be combined with both the *multi-label* and *hierarchical multi-label* formatting options. However, it is incompatible with *binary-label* since you cannot present multiple binary questions at once (that would be *multi-label*).
- *Multi-pass*, which combines the annotations of multiple workers for a single passage by partitioning the labels into groups (letting each worker focus on a small subset of labels at a time). This option is compatible with all format versions. To accomplish this with the *hierarchical multi-label* format, we simply partition the hierarchy into sub-trees using the top-level labels.
- *Hierarchical multi-pass*, in which there are different stages of annotation which determine whether child labels in the hierarchy need to be annotated. First, some worker is asked to annotate the passage with level-1 annotations. Based on their annotations, we create new tasks for any label the worker marked as positive. These new tasks are released in a second stage to annotate the child labels in level 2, and need not be labeled by the same worker as the level-1 labels. This option can be employed both in a *binary-label* setup, as well as in *multi-label*, but is incompatible with *hierarchical multi-label* formatting since it forces annotation to occur on distinct levels at a time.

3.4.3 Combinations. When combining the format options with pass-logic options, we get the following possible labeling schemes:

- **Single-pass multi-label** (single-pass multi) — A single worker annotates all level-2 labels at the same time, given in a flat list.
- **Single-pass hierarchical multi-label** (single-pass hrchl) — A single worker annotates all labels (level-1 and level-2) at the same time. They are shown the hierarchy in its entirety using hierarchical multi-label v1 formatting (see Figure 3).

- **Multi-pass binary-label** (multi-pass binary)[3] — The labels are given one by one to multiple workers, who annotate the passage in parallel for that single label. Annotations from all workers are then combined.
- **Multi-pass multi-label** (multi-pass multi) — Level-2 labels are partitioned into smaller groups, and these label-groups are then given to multiple workers who annotate the passage in parallel. Annotations from all workers are then combined.
- **Multi-pass hierarchical multi-label** (multi-pass hrchl) — The labels are partitioned according to level-1 labels. One worker will be given label 1 and its children 1.1, 1.2, ..., while another worker will be given label 2 and its children 2.1, 2.2, ... This pattern continues for all level-1 labels in the hierarchy. They are shown their section of the hierarchy using hierarchical multi-label v2 formatting (see Figure 3).
- **Hierarchical multi-pass binary-label** (hrchl-pass binary) — Level-1 labels are first given one by one to multiple workers, who annotate the passage in parallel for their single label. A second stage then looks at these annotations and determines which child labels need to be labeled (if a worker indicates a positive label for label 2, then we must annotate 2.1, 2.2, ..., else we can skip them). The second stage then gives the child-labels one by one to multiple workers in parallel, just like the first stage.
- **Hierarchical multi-pass multi-label** (hrchl-pass multi) — Level-1 labels are first given as a single group to one worker. A second stage then looks at this worker's annotations and determines which child labels need to be labeled (according to the same logic as in the binary case). The second stage then gives the child-labels in groupings according to their parent category (so a single worker will be given 2.1, 2.2, ... at once), just like in the first stage.

When partitioning the level-2 labels for *multi-pass multi*, we examine two possible choices: partitioning them using the groupings that already exist in the hierarchy, or partitioning them randomly into 5 groups (such that the number of groups is consistent with the other choice). We refer to these as *multi-pass **grouped** multi* and *multi-pass **random** multi*, respectively.

3.5 Controls

Beyond forcing an annotator training, we explore several additional controls. This section outlines the controls we took, and which factors we look at through post-hoc analysis. §3.5.1 looks at how we control cost, which is key to our experimental design.

Given the task difficulty, we limit access to the task to workers who (1) reside in the United States, (2) have completed at least 2,000 HITs, and (3) have a HIT approval percentage of above 99%, and (4) have a "Masters" qualification indicating they are workers that produce high quality annotations. These controls are facilitated by standard AMT tools, while most of the rest of the controls are implemented through our custom annotation platform.

We use a between-subjects design, meaning that we do not allow any worker to submit annotations for more than one labeling scheme. This avoids producing workers which are trained twice on the task. Further, the workers are not aware that there are multiple conditions. When publishing jobs on AMT we start with HITs that will send the workers to the first labeling scheme. Once we have collected enough annotations for this scheme, the current workers get blocked from beginning any new hits. The next labeling scheme then gets linked from the posted HITs, and new workers (which did not interact at all with the first labeling scheme) may begin annotation. This ensures workers are not aware of multiple schemes, even if they have seen the HIT advertised in their list of tasks previously. The description of the task is the same, except for the reward which fluctuates slightly to maintain a consistent budget (more details in §3.5.1), and we never inform the workers that there are multiple schemes. Workers are at most allowed to submit annotations for 200 unique passages.

We do not control exactly when these HITs are submitted to the AMT marketplace. Simply, we launch the next HITs shortly (1-2 hours) after gathering enough annotations for the last labeling scheme. When the last labeling scheme finishes collection during the night or late in the evening, we wait until the morning to launch the next scheme. One could argue that the populations of workers that click on tasks might vary meaningfully across the day. However, most annotations were collected during day-time in the United States, and we only allow workers from the United States. We also include this factor in our multiple regression analysis in §4.2, showing it does not significantly contribute to mean F1 score. Since we allow workers to complete any number of annotations they want (up to 200 unique passages), we cannot control how many workers are assigned to each condition. Instead, we allow annotation by new workers up until we have 3 copies of each of the 4,800 passage-label pairs. See Table 1 for more details.

interface design	≥ 1 tutorial Q	≥ full tutorial	≥ took exam	≥ 1 datapoint
hrchl multi-pass multi-label	87	76	54	33
multi-pass hrchl-label	57	37	35	29
multi-pass grouped multi-label	71	65	65	41
mutli-pass random multi-label	46	39	39	37
single-pass hrchl-label	27	8	8	7
single-pass multi-label	27	12	12	10

Table 1: Number of workers which began each stage of the data collection pipeline, broken down by labeling scheme. Since multi-pass schemes required more annotations, workers had more time to reserve HITs and submit annotations before it had been completed. This has some affect on our confidence intervals for single-pass schemes.

3.5.1 Cost. To control for cost, we approach the task from the perspective of a research team that wants to collect data to train an ML model.

Note that annotating text passages is very different from images. Whereas images can be cognitively processed near-instantaneously by a crowdsource worker, reading passages of text is more similar to the annotation of video or audio-clips. Performing a full read-through of the passage takes time, forcing a delay before selecting labels and thereby adding a significant temporal dimension to the annotation task. Therefore, we cannot compare the annotation of 10 passages with a single binary label to the annotation of 1 passage with 10 labels. In one case, the annotator has to spend 10 times

[3]Binary label schemes are not included in experiments due to their high cost.

more time reading than the other. This influences how to fairly pay workers. Instead, we must set a minimum reward threshold *per passage read-through* for workers, and consider the cost of data collection as variable. For a toy example of why cost would vary across labeling schemes, see Appendix L.

However, comparing labeling schemes without holding the total budget constant will not provide much value for ML researchers deciding which scheme to use. Research teams are heavily motivated by budgets, so how do they get the highest quality annotation for their money?[4] Answering this question is the focus of our experiments (§4). In particular, we set the reward per passage read-through for each labeling scheme to fully utilize the budget (ensuring that it was above a minimum of $0.10, which ensures that we are paying workers more than the United States minimum wage). This means in some labeling schemes the workers will get more rewards per passage than others, although these workers also have to consider more labels at the same time.

On AMT, workers are paid per HIT (one "unit" of work assigned to a worker) they complete. For each HIT, workers in our experiment will complete a small batch of passages (10-24 passages). Batching passages like this ensures the reward per HIT is not too small to attract workers. This also allows us to change the ratio of reward-to-number-of-passages, thereby controlling for the listed reward payment per HIT that workers see on the platform. We observe that changing this ratio (without changing the actual payment per work completed) causes noticeable differences in annotation throughput. This indicates a potentially large inefficiency in the AMT marketplace. For all labeling schemes, we launch the tasks with a ratio of reward-to-number-of-passages such that the reward is just above a dollar (as close to $1.01 as we can get while keeping the budget fixed). For all schemes we still keep the total budget spent constant for collecting the 14,400 labels needed (3 copies of 4,800 passage-label pairs).

3.5.2 Payment broken down by each labeling scheme. From the process described in 3.5.1, we then end up with the following rates of pay for each condition: The listed reward for *hrchl-pass multi* was $1.01 for 10 passages, all *multi-pass* options were $1.03 for 24 passages, and single-pass options were $2.16 for 10 passages (after first having tried $1.08 for 5 passages and finding throughput was too slow). The reward we give workers amounts to approximately $7-10 per hour (USD) as self-reported through TurkOpticon[5][23]. We do not have access to more granular hourly-rate estimates due to limitations with monitoring when workers are inactive (taking a break) versus when they are taking longer than usual time to read a question. However, we include analysis regarding distributions of time spent labeling each passage across the various schemes in Appendix D.

4 ANALYSIS

We collected three copies of annotations for each passage, for each of the 6 labeling schemes (§3.4.3) through AMT. In this section we compare the labeling schemes against each other on performance and examine the reasons for why performance varied across labeling schemes.

4.1 Performance Comparison

We evaluate the performance of workers against the ground-truth labels (§3.3). Majority labels are often computed to mitigate labeling error [52], but recent work has also shown the utility of high-quality individual annotations in order to estimate the distributions of human opinion [58]. The latter is particularly relevant in our setting where workers are labeling often subjective concerns: being able to measure the degrees of concern across individuals is relevant towards reducing vaccine hesitancy. We compute the precision, recall and F1 score for each label of the vaccine concerns taxonomy, and report an unweighted mean across the labels.

We employ a macro-level average of F1, which is computed by first finding the F1 score on every taxonomy label, and then averaging across all these labels. Note that in any analysis where we give an individual F1 score for each worker, the macro-averaging process happens in parallel for each worker. That is, the worker would be evaluated separately for each taxonomy label, and then an average performance is computed for that worker. However, for most of our analyses, we look at a single F1 score across all workers. In this case, we first pool all the annotations and treat them as if a single worker had submitted them. We then follow the macro-averaging process across the taxonomy labels. For further details on the metrics we use, see Appendix G.

We generate a choice/random baseline. For each passage, we draw 3 samples for each label from the binomial distribution with the probability p being determined by the gold-labeled data. We employ the same scheme to ensure consistency as described in Appendix F. Note that since F1 is computed using a macro average, and since there are "nans" in the data when positive labels are not generated, the mean F1 will not necessarily lie between the mean precision and mean recall.

interface design	prec	recall	F1
hrchl-pass multi	0.49	0.66	**0.56**
multi-pass hrchl	0.37	0.68	0.47
multi-pass grouped multi	0.41	**0.71**	0.50
multi-pass random multi	0.23	0.61	0.34
single-pass hrchl	**0.51**	0.56	0.52
single-pass multi	0.44	0.54	0.46
random baseline	0.06	0.06	0.13

Table 2: Mean performance of crowdsource workers against ground truth labels. Hrchl-pass multi-label performs best on mean F1. We include full breakdowns of these F1 scores (and other performance metrics) by label in Appendix N, as well as confidence intervals for all metrics in Appendix K

.

Workers annotating with *single-pass hrchl* had the highest precision of 0.51, while *multi-pass grouped multi* had the highest recall at 0.71. *Hrchl-pass multi* balanced these the best, with an F1 score of 0.56. Generally, the data indicates that single-pass options lead

[4]We should note that one possibility is to spend the same amount of money but vary the amount of data collected. This adds complexity to this question beyond the scope of this paper, since it would be necessary to build and evaluate ML model performances in order to measure the tradeoff of data *quantity*.

[5]The TurkOpticon page for our requester account shows 5/5 rating in Fairness and 5/5 rating Fast payments. This requester account was created solely for these experiments.

to higher precision, while multi-pass and hierarchical multi-pass options perform better on recall.

One possible explanation could be that when workers focus on a smaller set of labels, they have a lower chance of forgetting about them while reading the passage. The tradeoff would be that as workers see a longer list of labels, they have to be more certain the passage is speaking about a label to think of it and select it. It could also be possible that workers "want" to select *something* on each passage. When the options are few they tend to over-annotate, and when the options are many they find the obvious ones more easily, producing fewer false positives. There is some evidence for this explanation. The mean number of selections per passage in the single-pass schemes was 0.9, while the mean number of selections in multi-pass options was 1.5, indicating that partitioning the labels into smaller categories may cause workers to annotate more positives than if they are given all together.

4.2 Multiple Regression Analysis

We investigate the effects of various factors on worker F1 scores. In this section, we fit a multiple regression model to the F1 scores of each worker. See Table 1 for how many workers completed annotations in each labeling scheme. We consider several factors beyond the labeling scheme, including ones that were not controlled for in our experimental design (such as the time each labeling scheme was distributed on AMT) as well as factors which arise due to each worker's "luck": the percentage of passages they were given which were relatively easy, and how often they were shown a passage which *should* be labeled with some positive label.

The *labeling scheme* factor is a categorical variable encoding the 6 labeling schemes considered in this paper. *Multi-pass random multi* is set as the baseline for this analysis. *time started* is a variable encoding when during the day a given worker began annotating passages. It is given in seconds past midnight. *percentage easy/medium/hard/no agreement* factors encode the percentage of easy / medium / hard / or no agreement (referring to the 4 proxy levels for difficulty) labels which the given worker was presented with. Some workers, by luck, get easier or harder passage-label pairs shown to them, and here we hope to see what the effect of this is. Details on how we choose these 4 difficulty levels are given in §4.3.2 and Appendix M. *true pos freq* is a variable encoding what percentage of labels shown to a given worker *should* be labeled positively.

F1 scores were significantly improved by three labeling schemes above the baseline: multi-pass grouped multi-label (estimate = 0.13, p-value < 0.001), single-pass hrchl-label (estimate = 0.08, p-value < 0.05), and single-pass multi-label (estimate = 0.07, p-value < 0.1).

For factors beyond the labeling scheme, we see that the time of day each worker began the task did not have a statistically significant effect on the data (p-value = 0.66), whereas both the rate of true positives that workers encounter during annotation (estimate = 0.54, p-value < 0.001) and the percentage of easy passages they encounter (estimate = 0.39, p-value < 0.05) do have a statistical significance. This is especially of interest to us since these factors can be indirectly manipulated through the labeling scheme. We analyse these factors in further detail in §4.3.2 and §4.3.3.

Model factor	Estimate	95% CI	SE	p-value
labeling scheme (baseline = multi-pass random multi-label)				
hrchl-pass multi-label	0.0078	[-0.067, 0.083]	0.038	0.838
multi-pass grouped multi-label	0.1334	[0.057, 0.209]	0.039	0.001
multi-pass hrchl-label	-0.0257	[-0.096, 0.044]	0.036	0.471
single-pass hrchl-label	0.0821	[0.005, 0.160]	0.039	0.038
single-pass multi-label	0.0739	[-0.005, 0.153]	0.040	0.066
Additional numerical factors				
time started	-3e-7	[-2e-6, 1e-6]	7e-7	0.659
percentage easy	0.3929	[0.053, 0.733]	0.173	0.024
percentage medium	0.3518	[-0.079, 0.782]	0.219	0.109
percentage hard	-0.089	[-1.405, 1.227]	0.670	0.894
percentage no agreement	-0.505	[-1.411, 0.402]	0.461	0.274
true positive freq	0.5388	[0.321, 0.757]	0.111	0.000

Table 3: Multiple regression analysis for factors influencing worker F1 score on a per-label basis. easy / medium / hard / no agreement refer to the 4 categories described in §3.3. Note that a coefficient of 0.07 can indicate the difference between (for example) 0.50 and 0.57 F1 scores, since labeling scheme factors are coded as binary against the baseline scheme.

4.3 Contributing factors toward performance differences

In this section we perform deeper analysis on potential reasons for performance differences across labeling schemes.

4.3.1 Grouping labels. Overall, integrating the hierarchy into the labeling scheme seems to help with performance. One direct comparison we can make is between the two versions of *multi-pass multi-label* schemes. In one, *multi-pass random multi*, the level-2 labels are partitioned randomly and given to separate workers. In *multi-pass grouped multi*, we use the groupings that already exist due to the hierarchy. Comparing performance between these schemes helps us examine whether presenting conceptually similar categories together can boost performance.

In every single measurement (accuracy, precision, recall, F1) and in every single vote setting (sensitive, majority, unanimous), the grouped scheme outperforms random partitions. On individual workers' mean performance, *multi-pass grouped multi* scores 0.50 with a 95% confidence interval of $[0.45, 0.55]$, while *mutli-pass random multi* only scores 0.34 ($[0.29, 0.43]$). It seems important when partitioning the labels to group related labels together. It is unclear exactly why this is, but one possibility is that having the context of similar labels increases worker's understanding of the nuance between different cases. If they are shown a passage with a text which has criticism of research, it may be useful to be labeling both *"Issues with vaccine research → poor quality"* alongside *"Issues with vaccine research → lacking quantity"* rather than just one (without knowledge about the other).

4.3.2 Examining difficulty. Even though our task *generally* contains more ambiguity and is higher in cognitive load than other crowdsourced annotation tasks, there are of course easy cases to label. For example, the passage below (Figure 4) should very clearly be labeled with *"Health risks"*.

Pregnant Women Given Vaccine Have Babies with More Health Problems

Figure 4: An easy-to-label example passage from an anti-vaccination blog.

We utilize the ground-truth label categories discussed in §3.3, and examine the difference in performance as we vary difficulty. Importantly, we do not simply assign a difficulty measure to each passage, but rather to each passage-label pair. That means that we are able to mark that it is easy to annotate *"Health risks"* for the passage in Figure 4, but we can also mark that it is difficult to annotate the label *"High risk individuals"* if that was a label the authors did not immediately agree on. This analysis is done post-hoc. The passages are given at random ordering to workers, so workers will in expectation see the same proportion of difficult passages.

Figure 5: Worker mean F1 score versus increasingly difficult passage-label pairs. Note that some platforms perform better on more difficult passages, largely driven by their performance in precision. Note that the fourth category is excluded from this analysis due to lack of enough data, and semantic issues with how to compare performance against passages where both binary labels are technically valid.

Appendix M shows similar plots for accuracy, recall, and precision.

Focusing in on two comparable labeling schemes, the two single-pass versions, we see that performance on the labels diverges as difficulty increases. Performance on the easiest category (immediate agreement among authors) is almost identical (F1 score of 0.581 for *single-pass multi* and 0.582 for *single-pass hrchl*), while the difference is already +0.400 in the favor of *single-pass hrchl* as we reach the most difficult category where there is still author consensus. This generally supports the explanation that explicitly providing the hierarchy helps workers reason about difficult labels. It is unclear, however, exactly why the performance *increases* as difficulty increases for the *single-pass hrchl* scheme. It is possible that the tradeoff between the helpful structure and the harmful interface complexity interact such that this labeling scheme performs worse on easier passages. Alternatively, it may be an effect of correcting workers' priors for assigning a positive label.

4.3.3 True positive frequency. Beyond the interface design format shown to workers, and the pass-logic used to combine annotations, there may be other factors that impact their performance. Does a worker who sees lots of positive examples perform differently from a worker who rarely sees any positives?

The results of the multiple linear regression indicates that there is a significant increase in F1 score due to higher true positive frequencies shown to workers. Knowing this, we may want to design annotation platforms which "filter out" negative examples, so that more workers have higher true positive frequencies during annotation. See Appendix I for a plot of the relationship between true positive frequency and F1 score among all the workers. Intuitively, one reason higher true positive frequencies may cause better performance could be that workers *expect* to have to assign positive labels to some proportion of passages, which would cause them to over-assign positives.

We examine the performance differences between labels collected in *multi-pass grouped multi* and *hrchl-pass multi*. If we ignore the level-1 annotations collected in *hrchl-pass multi*, then the interface shown to workers in these two cases is identical. The only difference is which passages actually get shown. For *multi-pass grouped multi*, we show all available passages to the workers. There is no pre-filtering on relevant passages done. For *hrchl-pass multi*, we only show passages that already have a positive annotation of the parent label, meaning there is a high chance of more labels being relevant. In fact, the frequencies of true positives shown to workers jumps from 3% to 13% on average (a more than 4-fold increase) just from this pre-filtering. Below, we compare the label performance on the passages that were annotated in both schemes.

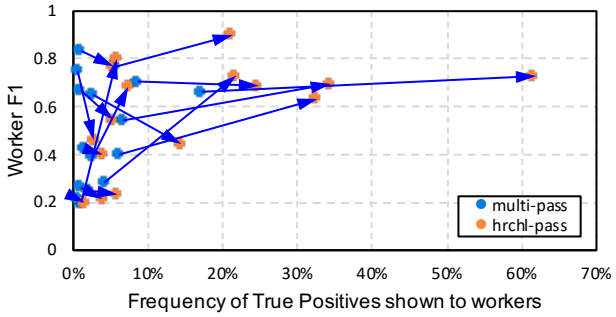

Figure 6: Worker F1 score as the frequency of seen true positives (TPs) varies. Labeling schemes determine which paragraphs get shown: all available (blue) or pre-filtered by the parent label (orange). Arrows link the same labels from one scheme to the other.

Note that overall we see a statistically significant positive correlation (Table 3), when we aggregate across all workers and examine changes on a per-label basis this trend is more nuanced. Overall, for passages directly annotated by workers in both schemes, *hrchl-pass multi* achieves a mean F1 score of 0.57 on level 2 labels whereas *multi-pass grouped multi* only scores 0.50. This is driven mainly by an improvement on precision (+6.7%) rather than recall, which stays fairly unaffected (+0.01). It therefore seems that better balancing class priors for the workers can help with their performance on the

task. This may warrant recommendations of a pre-filtering step to remove obvious true negatives. Ultimately, hierarchical multi-pass schemes acts as a form of pre-filtering, and seems to have a positive influence on worker performance.

4.4 Voting schemes

If one's primary goal is not to measure the distribution of judgments about a label, but rather to get a single binary answer for each passage, then employing a vote may still be beneficial. That is not the primary motivation of this work, but in order to give some guidance to the implications of our results for aggregation methods, we examine simple, threshold-based voting schemes.

We look at three possible vote setting to aggregate the three copies of annotations collected on each passage. In sensitive vote, only 1 positive vote (of 3) is required to mark a label as positive. In majority, 2 of 3 is required, and in unanimous all 3 must be positive to mark it positive.

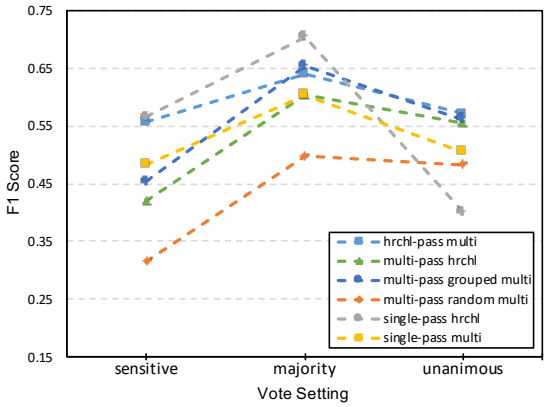

Figure 7: F1 score of each vote setting. Consecutive vote settings become increasingly conservative on positive labels, requiring 1, 2, or all 3 votes (respectively) to mark it positive. All labeling schemes maximize F1 score using majority score, indicating it is the safest choice for vote setting.

Majority vote balances precision and recall the best across all labeling schemes (see Figure 7). Note that as mentioned in §2, strong individual annotator performance may still be necessary for some tasks where aggregation is not possible. Majority vote also outperforms individual worker's performance in all labeling schemes, and the differences in performance between labeling schemes become less pronounced as you aggregate using voting. The highest F1 score was achieved by *single-pass hrchl* at 0.70 using majority vote. Sensitive vote scores the best on recall, with *multi-pass grouped multi* achieving 0.92, and unanimous vote settings perform the best on precision: *single-pass multi* scores 0.93. Increasing the vote threshold results in more conservative labeling, which is why we see an increase in precision.

5 DISCUSSION

There is a large body of work which has studied crowdsource annotations. How to make it scalable, how to keep quality high, and how to increase throughput. There is also a deep wealth of knowledge and best practices regarding user interface design. Much of this work, however, has been done for labeling tasks that are low-subjectivity, have clearly defined label definitions, and low cognitive load.

We studied six candidate labeling schemes in annotation of a taxonomy of vaccine concerns. Our motivation was to study whether the hierarchy itself could aid workers as they perform the annotation task, and how to design the annotation task for high-quality labels under a fixed budget. We found that integrating the hierarchy into the labeling scheme helps with improving annotation quality, whether explicitly in the interface or through logical passes made behind the scenes. Our analysis showed that a *hierarchical multi-pass multi-label* scheme performs best when considering individual worker performance. We believe individual worker performance to carry more importance when the tasks are inherently subjective, since a growing body of work is interested in predicting label distributions. Much like in [7] and [45], we find that workers assign more labels per passage on average when they are in multi-pass schemes versus single-pass ones.

However, if the priority is to collect high confidence labels rather than distributions of human opinion, we found that employing the *single-pass hierarchical multi-label* along with a majority vote achieves highest performance. Unlike [7] and [21], we don't see a drop-off in performance when using single-pass labeling schemes. Although we don't conduct a qualitative study, we did not receive notably different amounts of complaints from workers in any of the labeling schemes. Largely, complaints were not about the interface designs at all, but rather about being allowed to annotate more data after workers finished the batch or were banned for failing attention checks. When using majority vote, the choice of labeling scheme matters less than it does for individual worker performance. The ease of setup with single-pass options should not be undervalued either. Such labeling schemes are already supported natively in AMT's requester user interface, making it a strong option for smaller projects with a necessary quick turnaround.

Overall, we find that introducing the hierarchy helps almost universally across our experiments. In comparisons between partitioning the labels into groups randomly versus using the hierarchies structure, we find that using the hierarchy dominates across all performance metrics. Exposing the hierarchy explicitly helped performance on single-pass schemes by increasing worker performance on particularly difficult passages. We used a taxonomy specifically designed to achieve high agreement among crowdsource workers. While some previous work has indicated that integrating hierarchies into the data labeling process may harm performance ([51]), we find that it boosts it. The contexts here are different: our task is higher difficulty and therefore the hierarchy may aid in completing the task, but using a taxonomy specifically designed for high agreement among crowdsource annotations may also indicate that for these methods to work you may need a well-designed hierarchy.

5.1 Limitations

Use of Amazon Mechanical Turk. We conduct our experiments on AMT, one of the biggest and most popular crowdsourcing platforms. While AMT is similar to many other crowdsourcing platforms,

and while we do use a custom annotation platform which limits annotators interaction with AMT-specific UI, there are a few unique traits to AMT. First, the population of workers is hard to replicate to other platforms. We use several of AMT's built in qualifications to filter out workers, and there is no clear translation for which qualifications to use on other crowdsourcing platforms. Further, AMT has different payment expectations than other crowdsourcing communities. Some crowdsourcing platforms are purely volunteer based, while others attract short-time workers who complete only a few tasks. Ultimately, our choice to work on AMT is motivated by the size and popularity of this platform, thereby having results be relevant for a large set of researchers.

Omission of binary-label formats. Due to cost constraints, we could not experiment with labeling schemes which involved presenting binary choices to the workers. While this is representative of real-world scenarios for tasks similar to ours, it also leaves questions regarding whether or not the quality of annotation is potentially higher with these methods. However, we believe that given the vast cost differences of these methods, this choice is a reasonable assumption and will closely represent decisions made by the researchers for whom this analysis is intended.

Generalizing to new hierarchies. While we have no clear reason to expect our results are specific to the vaccine concerns hierarchy we used, we do not show or indicate that these results generalize well beyond it. For instance, as the size of the hierarchy grows, one might expect that the single-pass options become cognitively overbearing, and therefore multi-pass methods might begin increasing in relative performance. However, in offering useful analysis this is a choice that must be made.

Budget implications. We set a single budget and examine how to best optimize annotation performance against ground-truth labels on AMT. However, it may be the case that the best labeling scheme for our budget shows a less significant improvement when the budget is much higher and the reward given to workers is increased. Or there may be a different labeling scheme which performs better under a higher budget. One could imagine repeating our experiments at several budgets, and examining the relationship between a particular labeling scheme's data quality and the relative expense of data collection. Some schemes may be more cost efficient, showing small differences in worker performance across budgets, while others may only become viable at higher budgets. Unfortunately, running such experiments would make this research prohibitively expensive. The budget can be set based on previous experimentation regarding the minimum budget needed to achieve relatively high quality data, as well as confidently exceed the United States minimum wage. Teams that wish to collect data would likely avoid opting to pay more for the labels they are getting. For machine learning applications, it is well known that you may get more utility from collecting *additional* data, rather than increasing the quality of the labeled data [20].

6 CONCLUSION

We investigate various labeling schemes for crowdsourcing text annotation of difficult, high-subjectivity tasks and measure impact on worker performance against ground-truth labels. We find that integrating hierarchies into the labeling scheme helps with boosting performance.

Through analysis, we explore three potential indirect causes for improvement against ground-truth labels: (1) They group similar concepts together, improving F1 scores to 0.50 from 0.34 as compared to random groupings. (2) They allow relative increases in performance on difficult passages, leading to an increase in as much as +0.40 on F1 score on high difficulty examples. (3) They boost the true positive frequency, thereby increasing precision of workers without detriment to recall. We recommend considering incorporating hierarchies into the labeling process if optimizing for individual worker performance, while using a majority vote setting if solely optimizing for F1 score (achieving 0.70 with *single-pass hierarchical multi-label*).

ACKNOWLEDGMENTS

We thank Pardis Emami-Naeini and anonymous reviewers for feedback. This work was supported by NSF award IIS-2211526 and an award from Google.

REFERENCES

[1] Alberto Alemanno. 2018. How to counter fake news? A taxonomy of anti-fake news approaches. *European journal of risk regulation* 9, 1 (2018), 1–5.

[2] Fatma Arslan, Josue Caraballo, Damian Jimenez, and Chengkai Li. 2020. Modeling Factual Claims with Semantic Frames. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*. European Language Resources Association, Marseille, France, 2511–2520. https://aclanthology.org/2020.lrec-1.306

[3] Fatma Arslan, Naeemul Hassan, Chengkai Li, and Mark Tremayne. 2020. A Benchmark Dataset of Check-Worthy Factual Claims. *Proceedings of the International AAAI Conference on Web and Social Media* 14 (May 2020), 821–829. https://ojs.aaai.org/index.php/ICWSM/article/view/7346

[4] Natã M. Barbosa and Monchu Chen. 2019. Rehumanized Crowdsourcing: A Labeling Framework Addressing Bias and Ethics in Machine Learning. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3290605.3300773

[5] Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. https://doi.org/10.48550/arXiv.1508.05326 arXiv:1508.05326 [cs].

[6] Jonathan Bragg, Mausam, and Daniel Weld. 2013. Crowdsourcing Multi-Label Classification for Taxonomy Creation. *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing* 1 (Nov. 2013), 25–33. https://ojs.aaai.org/index.php/HCOMP/article/view/13091

[7] Mark Cartwright, Graham Dove, Ana Elisa Méndez Méndez, Juan P. Bello, and Oded Nov. 2019. Crowdsourcing Multi-label Audio Annotation Tasks with Citizen Scientists. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–11. https://doi.org/10.1145/3290605.3300522

[8] Xiangyu Chen, Yadong Mu, Shuicheng Yan, and Tat-Seng Chua. 2010. Efficient large-scale image annotation by probabilistic collaborative multi-label propagation. In *Proceedings of the 18th ACM international conference on Multimedia (MM '10)*. Association for Computing Machinery, New York, NY, USA, 35–44. https://doi.org/10.1145/1873951.1873959

[9] Lydia B. Chilton, Greg Little, Darren Edge, Daniel S. Weld, and James A. Landay. 2013. Cascade: crowdsourcing taxonomy creation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. Association for Computing Machinery, New York, NY, USA, 1999–2008. https://doi.org/10.1145/2470654.2466265

[10] Travis G. Coan, Constantine Boussalis, John Cook, and Mirjam O. Nanko. 2021. Computer-assisted classification of contrarian claims about climate change. *Scientific Reports* 11, 1 (Nov. 2021), 22320. https://doi.org/10.1038/s41598-021-01714-4 Number: 1 Publisher: Nature Publishing Group.

[11] A. P. Dawid and A. M. Skene. 1979. Maximum Likelihood Estimation of Observer Error-Rates Using the EM Algorithm. *Applied Statistics* 28, 1 (1979), 20. https://doi.org/10.2307/2346806

[12] Michaël Defferrard, Kirell Benzi, Pierre Vandergheynst, and Xavier Bresson. 2017. FMA: A Dataset For Music Analysis. https://doi.org/10.48550/arXiv.1612.01840 arXiv:1612.01840 [cs].

[13] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on*

Computer Vision and Pattern Recognition. 248–255. https://doi.org/10.1109/CVPR.2009.5206848

[14] Jia Deng, Olga Russakovsky, Jonathan Krause, Michael S. Bernstein, Alex Berg, and Li Fei-Fei. 2014. Scalable multi-label annotation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. Association for Computing Machinery, New York, NY, USA, 3099–3102. https://doi.org/10.1145/2556288.2557011

[15] Alexandra Eveleigh, Charlene Jennett, Ann Blandford, Philip Brohan, and Anna L. Cox. 2014. Designing for dabblers and deterring drop-outs in citizen science. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. Association for Computing Machinery, New York, NY, USA, 2985–2994. https://doi.org/10.1145/2556288.2557262

[16] Leah Findlater, Joan Zhang, Jon E. Froehlich, and Karyn Moffatt. 2017. Differences in Crowdsourced vs. Lab-based Mobile and Desktop Input Performance Data. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. Association for Computing Machinery, New York, NY, USA, 6813–6824. https://doi.org/10.1145/3025453.3025820

[17] Association for Computing Machinery. 2023. The 2012 ACM Computing Classification System. https://www.acm.org/publications/class-2012

[18] Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter. 2017. Audio Set: An ontology and human-labeled dataset for audio events. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 776–780. https://doi.org/10.1109/ICASSP.2017.7952261 ISSN: 2379-190X.

[19] Raafat George Saadé and Camille Alexandre Otrakji. 2007. First impressions last a lifetime: effect of interface type on disorientation and cognitive load. *Computers in Human Behavior* 23, 1 (Jan. 2007), 525–535. https://doi.org/10.1016/j.chb.2004.10.035

[20] Alon Halevy, Peter Norvig, and Fernando Pereira. 2009. The unreasonable effectiveness of data. *IEEE intelligent systems* 24, 2 (2009), 8–12.

[21] Eric Humphrey, Simon Durand, and Brian McFee. 2018. OpenMIC-2018: An Open Data-set for Multiple Instrument Recognition.. In *ISMIR*. 438–444.

[22] Ioanna Iacovides, Charlene Jennett, Cassandra Cornish-Trestrail, and Anna L. Cox. 2013. Do games attract or sustain engagement in citizen science? a study of volunteer motivations. In *CHI '13 Extended Abstracts on Human Factors in Computing Systems (CHI EA '13)*. Association for Computing Machinery, New York, NY, USA, 1101–1106. https://doi.org/10.1145/2468356.2468553

[23] Lilly C. Irani and M. Six Silberman. 2013. Turkopticon: interrupting worker invisibility in amazon mechanical turk. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. Association for Computing Machinery, New York, NY, USA, 611–620. https://doi.org/10.1145/2470654.2470742

[24] Robert M. Jacobson, Paul V. Targonski, and Gregory A. Poland. 2007. A taxonomy of reasoning flaws in the anti-vaccine movement. *Vaccine* 25, 16 (April 2007), 3146–3152. https://doi.org/10.1016/j.vaccine.2007.01.046

[25] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. 2017. The Kinetics Human Action Video Dataset. https://doi.org/10.48550/arXiv.1705.06950 arXiv:1705.06950 [cs].

[26] Ashish Khetan and Sewoong Oh. 2017. Achieving Budget-optimality with Adaptive Schemes in Crowdsourcing. http://arxiv.org/abs/1602.03481 arXiv:1602.03481 [cs, stat].

[27] Aniket Kittur, Ed H. Chi, and Bongwon Suh. 2008. Crowdsourcing user studies with Mechanical Turk. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '08)*. Association for Computing Machinery, New York, NY, USA, 453–456. https://doi.org/10.1145/1357054.1357127

[28] Aniket Kittur, Boris Smus, and Robert Kraut. 2011. CrowdForge: crowdsourcing complex work. In *CHI '11 Extended Abstracts on Human Factors in Computing Systems (CHI EA '11)*. Association for Computing Machinery, New York, NY, USA, 1801–1806. https://doi.org/10.1145/1979742.1979902

[29] Ranjay Krishna, Kenji Hata, Stephanie Chen, Joshua Kravitz, David A. Shamma, Li Fei-Fei, and Michael S. Bernstein. 2016. Embracing Error to Enable Rapid Crowdsourcing. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. 3167–3179. https://doi.org/10.1145/2858036.2858115 arXiv:1602.04506 [cs].

[30] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. 2020. The Open Images Dataset V4. *International Journal of Computer Vision* 128, 7 (July 2020), 1956–1981. https://doi.org/10.1007/s11263-020-01316-z

[31] Joey J. Lee, Eduard Matamoros, Rafael Kern, Jenna Marks, Christian de Luna, and William Jordan-Cooley. 2013. Greenify: fostering sustainable communities via gamification. In *CHI '13 Extended Abstracts on Human Factors in Computing Systems (CHI EA '13)*. Association for Computing Machinery, New York, NY, USA, 1497–1502. https://doi.org/10.1145/2468356.2468623

[32] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. 2015. Microsoft COCO: Common Objects in Context. https://doi.org/10.48550/arXiv.1405.0312 arXiv:1405.0312 [cs].

[33] Chris J. Lintott, Kevin Schawinski, Anze Slosar, Kate Land, Steven Bamford, Daniel Thomas, M. Jordan Raddick, Robert C. Nichol, Alex Szalay, Dan Andreescu, Phil Murray, and Jan van den Berg. 2008. Galaxy Zoo : Morphologies derived from visual inspection of galaxies from the Sloan Digital Sky Survey. *Monthly Notices of the Royal Astronomical Society* 389, 3 (Sept. 2008), 1179–1189. https://doi.org/10.1111/j.1365-2966.2008.13689.x arXiv:0804.4483 [astro-ph].

[34] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. 2015. Deep Learning Face Attributes in the Wild. In *2015 IEEE International Conference on Computer Vision (ICCV)*. 3730–3738. https://doi.org/10.1109/ICCV.2015.425 ISSN: 2380-7504.

[35] I. Scott MacKenzie and Colin Ware. 1993. Lag as a determinant of human performance in interactive systems. In *Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems (CHI '93)*. Association for Computing Machinery, New York, NY, USA, 488–493. https://doi.org/10.1145/169059.169431

[36] Akihiro Miyata, Yusaku Murayama, Akihiro Furuta, Kazuki Okugawa, Keihiro Ochiai, and Yuko Murayama. 2022. Gamification strategies to improve the motivation and performance in accessibility information collection. In *Extended Abstracts of the 2022 CHI Conference on Human Factors in Computing Systems (CHI EA '22)*. Association for Computing Machinery, New York, NY, USA, 1–7. https://doi.org/10.1145/3491101.3519783

[37] Meredith Ringel Morris, Jeffrey P. Bigham, Robin Brewer, Jonathan Bragg, Anand Kulkarni, Jessie Li, and Saiph Savage. 2017. Subcontracting Microwork. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. Association for Computing Machinery, New York, NY, USA, 1867–1876. https://doi.org/10.1145/3025453.3025687

[38] 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]* (Oct. 2020). http://arxiv.org/abs/2010.03532 arXiv: 2010.03532.

[39] May Honey Ohn and Khin-Maung Ohn. 2019. An evaluation study on gamified online learning experiences and its acceptance among medical students. *Tzu-Chi Medical Journal* 32, 2 (June 2019), 211–215. https://doi.org/10.4103/tcmj.tcmj_5_19

[40] Sharon Oviatt. 2006. Human-centered design meets cognitive load theory: designing interfaces that help people think. In *Proceedings of the 14th ACM international conference on Multimedia (MM '06)*. Association for Computing Machinery, New York, NY, USA, 871–880. https://doi.org/10.1145/1180639.1180831

[41] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ Questions for Machine Comprehension of Text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Austin, Texas, 2383–2392. https://doi.org/10.18653/v1/D16-1264

[42] Joni O. Salminen, Hind Almerekhi, Milica Milenkovic, Soon-Gyo Jung, Jisun An, Haewoon Kwak, and Bernard Jim Jansen. 2018. Anatomy of Online Hate: Developing a Taxonomy and Machine Learning Models for Identifying and Classifying Hate in Online News Media. In *International Conference on Web and Social Media*.

[43] Eunjin Seong and Seungjun Kim. 2020. Designing a Crowdsourcing System for the Elderly: A Gamified Approach to Speech Collection. In *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems (CHI EA '20)*. Association for Computing Machinery, New York, NY, USA, 1–9. https://doi.org/10.1145/3334480.3382999

[44] Nihar B. Shah, Sivaraman Balakrishnan, and Martin J. Wainwright. 2021. A Permutation-based Model for Crowd Labeling: Optimal Estimation and Robustness. *IEEE Transactions on Information Theory* 67, 6 (June 2021), 4162–4184. https://doi.org/10.1109/TIT.2020.3045613 arXiv:1606.09632 [cs, math, stat].

[45] Gunnar A. Sigurdsson, Olga Russakovsky, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. 2016. Much Ado About Time: Exhaustive Annotation of Temporal Data. https://doi.org/10.48550/arXiv.1607.07429 arXiv:1607.07429 [cs].

[46] Rickard Stureborg, Bhuwan Dhingra, Jun Yang, and Lavanya Vasudevan. 2023. Development and validation of VaxConcerns: a taxonomy for vaccine concerns with crowdsource-viability. http://rickard.stureborg.com/papers/vax_taxonomy

[47] Alexandra Swanson, Margaret Kosmala, Chris Lintott, Robert Simpson, Arfon Smith, and Craig Packer. 2015. Snapshot Serengeti, high-frequency annotated camera trap images of 40 mammalian species in an African savanna. *Scientific Data* 2, 1 (June 2015), 150026. https://doi.org/10.1038/sdata.2015.26 Number: 1 Publisher: Nature Publishing Group.

[48] John Sweller. 2011. CHAPTER TWO - Cognitive Load Theory. In *Psychology of Learning and Motivation*, Jose P. Mestre and Brian H. Ross (Eds.). Vol. 55. Academic Press, 37–76. https://doi.org/10.1016/B978-0-12-387691-1.00002-8

[49] James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for Fact Extraction and VERification. *arXiv:1803.05355 [cs]* (Dec. 2018). http://arxiv.org/abs/1803.05355 arXiv:1803.05355.

[50] Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. 2018. The inaturalist species classification and detection dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 8769–8778.

[51] Carl Vondrick, Donald Patterson, and Deva Ramanan. 2013. Efficiently Scaling up Crowdsourced Video Annotation. *International Journal of Computer Vision* 101, 1 (Jan. 2013), 184–204. https://doi.org/10.1007/s11263-012-0564-1
[52] Jacob Whitehill, Ting-fan Wu, Jacob Bergsma, Javier Movellan, and Paul Ruvolo. 2009. Whose Vote Should Count More: Optimal Integration of Labels from Labelers of Unknown Expertise. In *Advances in Neural Information Processing Systems*, Vol. 22. Curran Associates, Inc. https://papers.nips.cc/paper/2009/hash/f899139df5e1059396431415e770c6dd-Abstract.html
[53] Wayne A. Wickelgren. 1977. Speed-accuracy tradeoff and information processing dynamics. *Acta Psychologica* 41, 1 (Feb. 1977), 67–85. https://doi.org/10.1016/0001-6918(77)90012-9
[54] 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)* (New Orleans, Louisiana). Association for Computational Linguistics, 1112–1122. http://aclweb.org/anthology/N18-1101
[55] Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, and Jianxiong Xiao. 2016. LSUN: Construction of a Large-scale Image Dataset using Deep Learning with Humans in the Loop. https://doi.org/10.48550/arXiv.1506.03365 arXiv:1506.03365 [cs] version: 3.
[56] Jingpu Zhang, Zuping Zhang, Zixiang Wang, Yuting Liu, and Lei Deng. 2018. Ontological function annotation of long non-coding RNAs through hierarchical multi-label classification. *Bioinformatics* 34, 10 (May 2018), 1750–1757. https://doi.org/10.1093/bioinformatics/btx833
[57] Mingrui Ray Zhang, Shumin Zhai, and Jacob O. Wobbrock. 2019. Text Entry Throughput: Towards Unifying Speed and Accuracy in a Single Performance Metric. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3290605.3300866
[58] Xiang Zhou, Yixin Nie, and Mohit Bansal. 2021. Distributed NLI: Learning to Predict Human Opinion Distributions for Language Reasoning. *arXiv:2104.08676 [cs]* (April 2021). http://arxiv.org/abs/2104.08676 arXiv: 2104.08676.

A THE VACCINE CONCERNS (VAXCONCERNS) TAXONOMY

Concern	Rationale
1. Issues with Research	1.1 Lacking Quantity
	1.2 Poor Quality
	1.3 Fallible science
2. Lack of Benefits	2.1 Imperfect protection
	2.2 Herd immunity
	2.3 Natural immunity
	2.4 Insufficient risk
	2.5 Existing alternatives
3. Health Risks	3.1 Direct transmission
	3.2 Harmful ingredients
	3.3 Specific side effects
	3.4 Dangerous delivery
	3.5 High-risk individuals
4. Disregard of Individual Rights	4.1 Religious and Ethical Beliefs
	4.2 Right to Autonomy
5. Untrustworthy Actors	5.1 Incompetence
	5.2 Profit motives
	5.3 Censorship
	5.4 Conspiracy

Figure 8: "VaxConcerns" taxonomy used to label all passages in the experiments

B ALTERNATIVE TEXT FOR FIGURES

Figure 1: "Three diagrams are shown describing single-pass, multi-pass, and hierarchical multi-pass routing logic. For single-pass, all set of labels are given to one worker. For multi-pass, the labels are partitioned into groups (1,2,...) and given to separate workers (A,B,...). For hierarchical multi-label, the top level labels are given to one worker, who's annotations determine whether or not the child labels will be given as a new group to annotators downstream. This example shows the case where the first worker labels TFFTF, and downstream there are two tasks set up for new workers to label the children of label 1 and label 4, respectively." **Figure 2:** "The figure shows an example passage that reads: 'The minister of fear (the CDC) was working overtime peddling doom and gloom, knowing that frightened people do not make rational decisions — nothing sells vaccines like panic.'" **Figure 3:** "Four diagrams are shown side by side. In each diagram there are a set of checkboxes or radio buttons indicating how the labels will be presented to the user. Binary label (the leftmost diagram) contains a simple question '1.1?' and below it a radio button reading 'yes' or 'no'. Multi-label contains a simple list of checkboxes labeled '1.1, 1.2, ...'. Hierarchical multi-label v1 contains staggered checkboxes where the leftmost checkboxes read '1, 2' and the boxes immediately under these are tucked under them, reading '1.1, ...' for the parent label '1', and '2.1, ...' for the parent label '2'. Hierarchical multi-label v2 contains both the radio button setup from the leftmost diagram, as well as the checkboxes from multi-label underneath them." **Figure 4:** "A table is shown with column headers reading 'interface design', 'greater than or equal to 1 tutorial Q', 'greater than or equal to full tutorial', 'greater than or equal to took exam', and 'greater than or equal to 1 datapoint'. The table shows values for all 6 labeling schemes." **Figure 5:** "A table is shown with values for precision, recall, and F1 score. These metrics are given for each of the 6 labeling schemes, and a random baseline is shown at the bottom. Hrchl-pass multi has bold font at the f1 score indicating it is the highest value in that column: 0.56." **Figure 6:** "A two-part table is shown with column headers 'model factor', 'estimate', '95% CI', 'SE', and 'p-value'. The first part of the table (top) has a subheading that reads 'labeling_scheme (baseline=multi-pass random multi-label)', and the second part of the table (bottom) has a subheading that reads 'additional numerical factors'. The first part includes 5 of the 6 labeling schemes, while the bottom includes new factors such as 'time_started', 'percentage_easy', and 'true_positive_freq'. Some values in the table are denoted '*' which represents a p-value below 0.001." **Figure 7:** "The figure shows an example passage that reads: 'Pregnant women given vaccine have babies with more health problems'" **Figure 8:** "A bar chart is shown giving the value for worker F1 score on the X axis ranging from 0 to 1, and the difficulty on the Y axis. The labels, from top to bottom, on the Y axis read 'immediate author agreement', 'author agreement after providing rationale', and 'author agreement after discussion'." **Figure 9:** "A scatter plot shows orange and blue dots generally following a linearly positive relationship. The orange dots are labeled 'hrchl-pass' and the blue dots 'multi-pass'. On the X axis: 'Frequency of True Positives shown to workers'. On the Y axis: Worker F1. Each blue dot is paired with an orange dot through an arrow which is drawn between them pointing towards the orange dot." **Figure 10:** "A line plot is shown

with dashed lines between three dots. The dots are lined up at tickmarks labeled 'sensitive', 'majority', and 'unanimous'. This is repeated for all 6 labeling schemes, leaving 6 connected dotted lines all in different colors. Every one of the lines follows an inverted V shape, with their highest point being over the 'majority' tick mark."

C DEGRADATION IN WORKER TASK PERFORMANCE OVER TIME

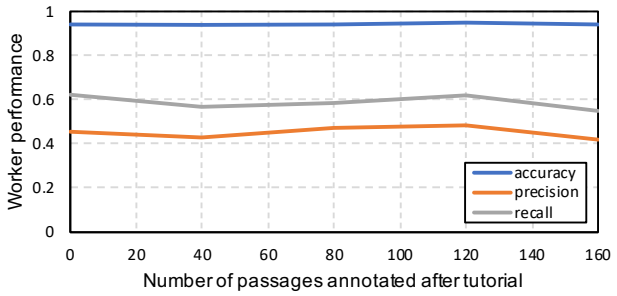

Figure 9: Mean performance of workers as they annotate passages. Worker performance fluctuates very little as they gain experience on the platform.

D DISTRIBUTIONS OF TIME SPENT LABELING EACH PASSAGE

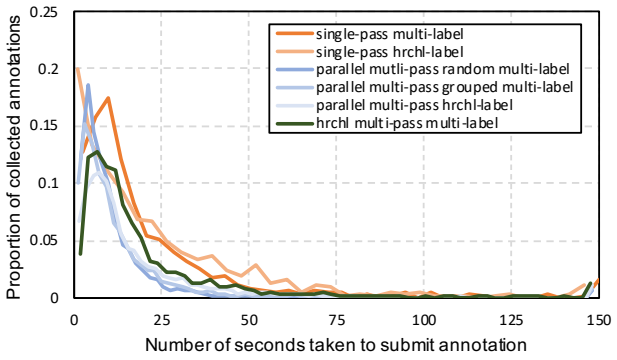

Figure 10: Distributions for amount of time spent labeling each passage, broken down by each interface design. Values are limited at a maximum of 2.5 minutes to account for behavior such as stepping away from the task to take a break.

E SCREENSHOT OF DEFINITIONS TASK IN ANNOTATION PLATFORM

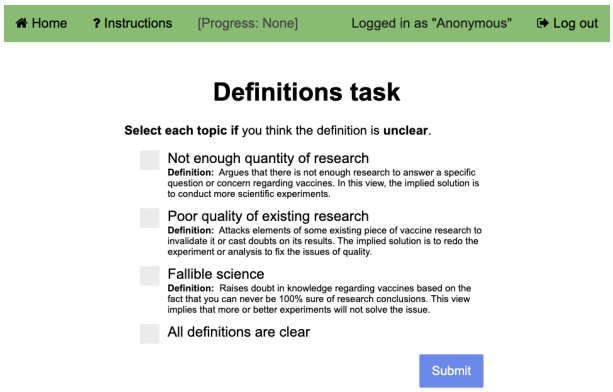

Figure 11: Definitions task presented to workers before they annotate any passages. Note that we are using a custom annotation platform to achieve a higher level control than offered by AMT with respect to quality checks, batching multiple passages for a single hit, and measuring worker behaviors, among other things.

F METHODS FOR PRE-PROCESSING DATA

We ignore all data collected for tutorial paragraphs, entrance exam, and quality checks when assessing workers. In the future we may assess how to leverage any information about worker performance on the tutorial towards improving data collection quality.

We remove all data collected from a worker if they failed the entrance exam or a quality check. Such workers were also banned in live time during data collection to avoid spending any further of our research budget on their data collection. To be clear, when setting the budgets for each labeling scheme, we do not factor in additional cost due to such workers. Rather, we re-annotate all the passages which they had been paid for.

For certain labeling schemes, consistency between level 1 and level 2 is forced by the annotation platform. For example, in single-pass hrchl-label, workers cannot submit a positive value for a level 2 label without also submitting a positive value for its parent node in the taxonomy. This is achieved through some javascript in the annotation platform and is visually confirmed to the workers while labeling. However, for other labeling schemes such as multi-pass multi-label, the separation of L1 and L2 labels into separate screens leads to collecting data which is not necessarily consistent. Since any real-world use of this latter type of labeling scheme would include corrections for consistency, we correct for this during post-processing. This ensures fair comparison between the labeling schemes such that we don't disadvantage multi-pass schemes. To further ensure comparison is consistent, we make sure to look at performance on level 2 labels on its own during our analysis.

G METRICS

We measure annotator performance using the F1 score, a harmonic mean of precision and recall. This score is commonly used in machine learning tasks and is a well understood and cared about metric in research communities such as natural language processing (NLP). Examining this metric gives more utility for NLP researchers, since the positively labeled examples will be the most informative during model training. One can achieve very high accuracy simply by marking all passages as negative. Since each worker's performance can be evaluated across each label in the taxonomy individually, we must use some aggregation technique when presenting our scores. We employ a macro-level average of F1, which is computed by first finding the F1 score on every taxonomy label, and then averaging across all these labels. In the case where we give an F1 score for each worker, the macro-averaging process happens in parallel for each worker. In the case where we give a single F1 score for all workers, we take all the annotations and treat them as if a single worker had filled out all annotations and we then follow the macro-averaging process. This method of averaging is preferable in our setting as opposed to a samples-based average which would compute F1 over a single passage, and then aggregate across passages. Semantically, we use this method since we also care about worker performance on each individual label and can thereby inspect these values. Inspecting worker performance on each passage is less important to us, since the set of passages used are meant to be a sample of the types of passages annotated in any application of our work.

H TASK UPTAKE

We train each worker before they complete any real annotations. Workers get paid for this training in order to ensure a fair and non-predatory ([4]) employment. This leads to additional costs to those interested in paying for the data labeling, since some workers can go through some or even all of the training, yet submit no actual annotations. Such workers never become "productive." Below we report the task uptake, that is the percentage of workers who became productive, out of all the workers which completed at least one tutorial example. We also report an inefficiency number, which is the percentage of extraneous annotations collected from training and attention checks (discussed in detail in §3.2). The inefficiency number compares the extraneous annotations to the total number of "useful" annotations collected. For multi-pass schemes, the workers were allowed to complete one partition (group) of the labels and then begin a new partition. This occurred seamlessly, whereas there was a multi-day delay for the *hrchl-pass multi* scheme, meaning that there were less return workers for this task and thus decreasing the total task uptake.[6]

Task inefficiency (which is directly proportional to additional incurred cost) is lowest for *hrchl-pass multi*. The task uptake is the highest for *multi-pass random multi* at 80%, which contrasts the performance trends shown in §4.1. One potential explanation for this could be that annotators underestimate the difficulty of the task when they're presented randomly partitioned labels.

[6]This multi-day delay was due to the annotation platform we used not supporting this immediate switch when the first labeling scheme was deployed.

interface design	Task uptake	Inefficiency
hrchl-pass multi	38%	**28%**
multi-pass hrchl	51%	41%
multi-pass grouped multi	58%	56%
multi-pass random multi	**80%**	38%
single-pass hrchl	26%	42%
single-pass multi	37%	51%

Table 4: Data collection inefficiencies due to training examples completed by workers. Uptake shows the percentage of workers who complete at least one "real" passage, while inefficiency shows the total percentage of extraneous annotations collected. Inefficiency is the lowest using a hrchl-pass scheme

I TRUE POSITIVE FREQUENCY VS F1 SCORES

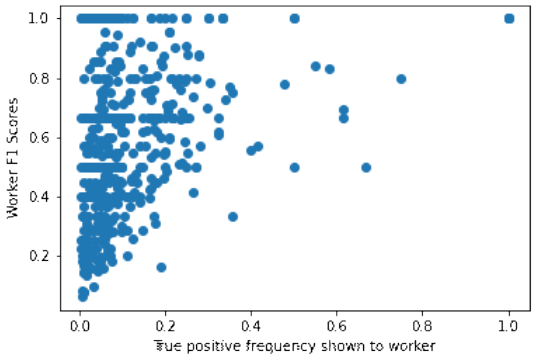

Figure 12: Relationship between the true positive frequency of each worker and their F1 scores.

J WORKER AGREEMENT ON LABELS BY LABELING SCHEME

	Mean Scott's Pi		
interface design	All Labels	Level 1	Level 2
hrchl multi-pass multi-label	**0.401**	0.336	**0.419**
parallel multi-pass hrchl-label	0.375	**0.412**	0.365
parallel multi-pass grouped multi-label	0.328	-	0.328
parallel mutli-pass random multi-label	0.378	-	0.378
single-pass hrchl-label	0.292	0.343	0.276
single-pass multi-label	0.294	-	0.294

Table 5: Mean Scott's Pi label agreement between crowdsource workers in each labeling scheme

K BOOTSTRAP CONFIDENCE INTERVALS

To find the confidence intervals included in any results of the paper, we use bootstrap confidence intervals. This is done directly on the raw data we collected, where a datapoint is a single submission by a worker. That is, for single-pass schemes the datapoint will include

all labels from the taxonomy, but for multi-pass schemes the datapoint will only include a subset of the labels. We draw N=10,000 samples with replacement from the original data, then find the performance metric using the process outlined in the paper (including all preprocessing). We then use these 10,000 measurements of each performance metric to compute 99% confidence intervals.

We do this sampling on the subset of annotations which ultimately contribute to the performance metric, rather than all the annotations we receive. This ensures we are not sampling (for example) tutorial annotations which we will ignore in the final calculations anyway.

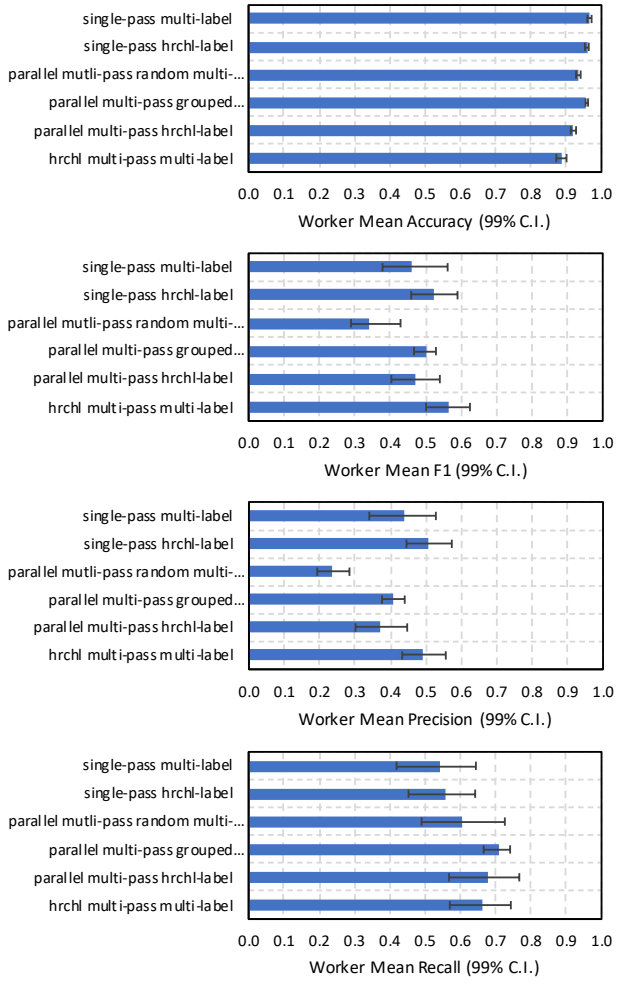

Figure 13: Confidence intervals for worker mean performance metrics.

L TOY EXAMPLE OF ANNOTATION COSTS

Suppose the team wants to collect approximately 10,000 fully labeled passages in order to provide high-quality training data. How much will it cost them to use each labeling scheme?

If the team assumes the longest part of labeling is due to reading, and want to guarantee a strong hourly wage for workers, then they can fix the reward at $0.10 per passage (this is a toy example, but loosely this is in line with paying above minimum wage in the United States). Then, the cost of data collection has to do with how many times they ask workers to read a single passage, just to collect a full set of labels. If chunking the question into smaller subsets (multi-pass), the cost will be greater. The extreme is when you ask a new worker to read the passage once for every label in the taxonomy. See Figure 14 for a breakdown of the cost to carry out this annotation.

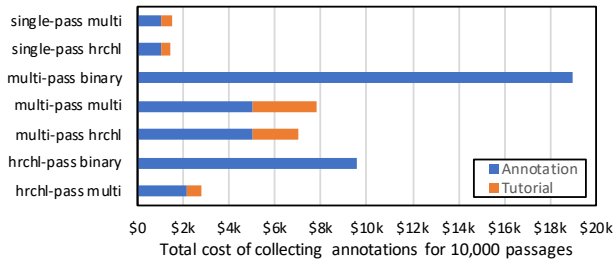

Figure 14: Cost for labeling 10,000 passages under various labeling schemes, assuming a reward of $0.10 per passage. Orange bars show the cost of training workers (further details in §H). Note that we do not show orange bars for the binary-label setups since we were not able to run these experiments and estimate the average overhead of training workers. Despite this, binary labeling is prohibitively expensive: 8.8x the cost of a hrchl-pass multi-label scheme.

We see that binary labeling schemes are prohibitively expensive. For this reason we do not include binary labeling schemes in further comparisons.

Overall, single-pass options are the cheapest (both below $2,000), including the cost of training the workers (paying them for completing the tutorial examples). Multi-pass options have a higher range, $5,000 – 8,000. *Hrchl-pass multi* balances cost (under $3,000) but still uses multiple workers to complete one passage's annotations.

M DETAILS FOR PRODUCING A PROXY FOR DIFFICULTY

Instant agreement ("easy") are the passage-label pairs for which the entire lab (3 authors plus 3 more students) gave the same value while individually annotating. During this step, the team looked up any terms we were unfamiliar or unclear about, just like the AMT workers are instructed to do.

Agreement after writing rationales ("medium"): highlight specific parts of the passage and justify why a label should or shouldn't apply. We only went through this process for passages where we had disagreed in the first step. During this process, we include any level 1 label if there is disagreement within any of its children, even if all team members agreed on the level 1 label.

For the remaining disagreement, we had a brief (1-2 minute) discussion for every passage-label pair, reading everyone's given rationale to see whether one of us could convince the others. This process included both reading the rationales written in 1 as well as generating new rationales (in discussion). The passage-label pairs agreed upon in this stage are referred to as "hard".

See Figure 15 for a few plots examining the worker performance metrics as each difficulty category varies.

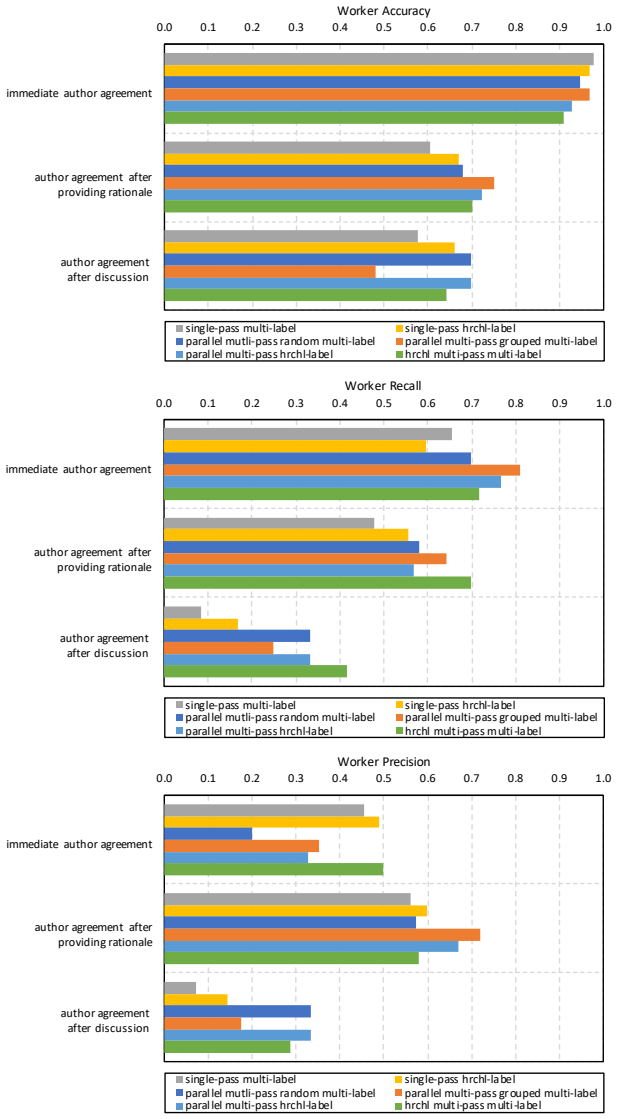

Figure 15: Worker mean performance metrics for each labeling scheme as difficulty increases (going down). Note that recall decreases steadily for all schemes, while precision largely gets better in the medium category, before experiencing a large dropoff again. Accuracy decreases steadily for most platforms. Some platforms have larger drop-offs in performance than others, possibly explaining the difference in F1 score.

The remaining passages are marked using each of our individual votes, and placed aside ("no agreement"). We consider these passages to be too subjective to give a gold label for, and don't evaluate workers on them since any label could be valid so long as they have a strong justification. In future work we may consider looking at the justification/rationale of an AMT worker to assess their performance on highly subjective passage-label pairs.

We make note of which category each passage-label pair was resolved in, such that we can perform an analysis into how the "difficulty" of passages affect labeling performance. We recognize that this is not necessarily a direct measurement of how difficult it is to label a passage, but we make the assumption that any passage that requires increasingly more thought or discussion to reach consensus will imply this passage is more difficult.

N PERFORMANCE BY TAXONOMY LABEL

See Tables in Figure 16 for each labeling scheme detailing the performance broken down by each individual label of the taxonomy. Final scores are computed for each metric by taking averages across these taxonomy labels.

hrchl multi-pass multi-label

Label	acc	prec	recall	f1
1	**0.91**	**0.54**	**0.39**	**0.45**
1.1	0.87	0.68	0.79	0.73
1.2	0.74	0.58	0.69	0.63
1.3	0.89	0.00	nan	nan
2	**0.94**	**0.72**	**0.48**	**0.58**
2.1	0.82	0.80	0.62	0.70
2.3	0.94	0.30	1.00	0.46
2.4	0.96	0.60	0.50	0.55
2.5	0.97	0.71	0.83	0.77
3	**0.88**	**0.87**	**0.57**	**0.69**
3.1	0.99	0.00	nan	nan
3.2	0.96	0.96	0.84	0.90
3.3	0.72	0.87	0.63	0.73
3.4	0.92	0.29	0.67	0.40
3.5	0.85	0.14	0.50	0.21
4	**0.94**	**0.42**	**0.43**	**0.43**
4.1	0.97	0.67	1.00	0.80
4.2	0.74	0.32	0.71	0.44
5	**0.89**	**0.74**	**0.46**	**0.57**
5.1	0.92	0.12	0.67	0.20
5.2	0.86	0.76	0.63	0.69
5.3	0.94	0.57	0.87	0.68
5.4	0.78	0.15	0.58	0.23
Mean	**0.89**	**0.49**	**0.66**	**0.56**

parallel multi-pass hrchl-label

Label	acc	prec	recall	f1
1	**0.74**	**0.24**	**0.81**	**0.37**
1.1	0.95	0.44	0.75	0.55
1.2	0.89	0.30	0.64	0.41
1.3	0.97	0.00	nan	nan
2	**0.67**	**0.19**	**0.80**	**0.30**
2.1	0.95	0.65	0.62	0.63
2.3	1.00	0.75	1.00	0.86
2.4	0.99	0.43	0.50	0.46
2.5	1.00	0.83	0.83	0.83
3	**0.89**	**0.72**	**0.87**	**0.79**
3.1	0.99	0.00	nan	nan
3.2	0.97	0.67	0.73	0.70
3.3	0.92	0.86	0.63	0.73
3.4	0.96	0.14	0.38	0.21
3.5	0.97	0.19	0.67	0.30
4	**0.82**	**0.16**	**0.63**	**0.26**
4.1	0.94	0.11	0.67	0.20
4.2	0.96	0.37	0.67	0.48
5	**0.75**	**0.36**	**0.73**	**0.48**
5.1	0.95	0.06	0.67	0.11
5.2	0.94	0.86	0.37	0.52
5.3	0.96	0.33	0.47	0.39
5.4	0.94	0.23	0.83	0.36
Mean	**0.92**	**0.37**	**0.68**	**0.47**

parallel multi-pass grouped multi-label

Label	acc	prec	recall	f1
1	-	-	-	-
1.1	0.91	0.20	0.46	0.28
1.2	0.93	0.41	0.39	0.40
1.3	0.93	0.00	nan	nan
2	-	-	-	-
2.1	0.95	0.61	0.49	0.54
2.3	1.00	0.60	1.00	0.75
2.4	0.99	0.56	0.83	0.67
2.5	1.00	0.83	0.83	0.83
3	-	-	-	-
3.1	1.00	nan	nan	nan
3.2	0.97	0.70	0.85	0.77
3.3	0.90	0.82	0.55	0.66
3.4	0.97	0.29	0.78	0.42
3.5	0.97	0.18	0.50	0.26
4	-	-	-	-
4.1	0.93	0.11	0.83	0.20
4.2	0.97	0.50	0.93	0.65
5	-	-	-	-
5.1	0.96	0.12	1.00	0.21
5.2	0.96	0.88	0.59	0.71
5.3	0.96	0.33	0.47	0.39
5.4	0.90	0.15	0.83	0.25
Mean	**0.96**	**0.41**	**0.71**	**0.50**

parallel mutli-pass random multi-label

Label	acc	prec	recall	f1
1	-	-	-	-
1.1	0.93	0.27	0.42	0.33
1.2	0.88	0.27	0.56	0.36
1.3	0.92	0.00	nan	nan
2	-	-	-	-
2.1	0.94	0.54	0.67	0.60
2.2	0.99	0.00	nan	nan
2.3	0.98	0.18	1.00	0.30
2.4	0.97	0.11	0.33	0.17
2.5	0.99	0.45	0.83	0.59
3	-	-	-	-
3.1	0.99	0.00	nan	nan
3.2	0.92	0.38	0.64	0.47
3.3	0.91	0.74	0.72	0.73
3.4	0.93	0.05	0.22	0.09
3.5	0.89	0.06	0.67	0.11
4	-	-	-	-
4.1	0.92	0.12	1.00	0.21
4.2	0.94	0.25	0.67	0.36
5	-	-	-	-
5.1	0.91	0.02	0.33	0.04
5.2	0.94	0.70	0.55	0.62
5.3	0.94	0.24	0.53	0.33
5.4	0.86	0.08	0.58	0.14
Mean	**0.93**	**0.23**	**0.61**	**0.34**

single-pass hrchl-label

Label	acc	prec	recall	f1
1	**0.92**	**0.58**	**0.46**	**0.51**
1.1	0.96	0.54	0.29	0.38
1.2	0.95	0.63	0.42	0.50
1.3	0.98	0.00	nan	nan
2	**0.96**	**0.94**	**0.63**	**0.76**
2.1	0.97	0.95	0.54	0.69
2.2	1.00	nan	nan	nan
2.3	1.00	0.50	0.67	0.57
2.4	0.99	0.57	0.67	0.62
2.5	1.00	1.00	0.67	0.80
3	**0.89**	**0.81**	**0.70**	**0.75**
3.10	1.00	0.00	nan	nan
3.20	0.97	0.77	0.70	0.73
3.30	0.90	0.78	0.61	0.68
3.40	0.97	0.26	0.56	0.36
3.50	0.96	0.11	0.50	0.18
4	**0.96**	**0.57**	**0.53**	**0.55**
4.1	0.96	0.15	0.67	0.24
4.2	0.98	nan	0.00	nan
5	**0.88**	**0.62**	**0.64**	**0.63**
5.1	0.98	0.18	1.00	0.30
5.2	0.95	0.80	0.55	0.65
5.3	0.96	0.28	0.47	0.35
5.4	0.94	0.17	0.50	0.25
Mean	**0.96**	**0.51**	**0.56**	**0.52**

single-pass multi-label

Label	acc	prec	recall	f1
1	-	-	-	-
1.1	0.96	0.50	0.29	0.37
1.2	0.95	0.62	0.36	0.46
1.3	0.99	0.00	nan	nan
2	-	-	-	-
2.1	0.97	0.95	0.49	0.64
2.2	1.00	0.00	nan	nan
2.3	1.00	0.50	1.00	0.67
2.4	0.99	0.75	0.50	0.60
2.5	0.99	1.00	0.33	0.50
3	-	-	-	-
3.10	1.00	0.00	nan	nan
3.20	0.95	0.55	0.55	0.55
3.30	0.89	0.81	0.47	0.60
3.40	0.97	0.27	0.44	0.33
3.50	0.96	0.09	0.33	0.14
4	-	-	-	-
4.1	0.99	0.42	0.83	0.56
4.2	0.98	0.53	0.67	0.59
5	-	-	-	-
5.1	0.97	0.13	1.00	0.22
5.2	0.96	0.88	0.55	0.67
5.3	0.95	0.23	0.40	0.29
5.4	0.91	0.12	0.50	0.19
Mean	**0.97**	**0.44**	**0.54**	**0.46**

Figure 16: Performance breakdown by taxonomy label for each labeling scheme. Note that nan values appear when there is no positive label given. No positive passages in the ground-truth mean we cannot compute a recall, and no positive passages in the worker annotations mean we cannot compute precision.