DynamicLabels: Supporting Informed Construction of Machine Learning Label Sets with Crowd Feedback

Jeongeon Park[*]
jeongeon.park@kaist.ac.kr
Electrical Engineering and Computer Science, DGIST
Daegu, Republic of Korea

Eun-Young Ko
eunyoungko@kaist.ac.kr
School of Computing, KAIST
Daejeon, Republic of Korea

Yeon Su Park
yeonsupark@kaist.ac.kr
School of Computing, KAIST
Daejeon, Republic of Korea

Jinyeong Yim[†]
jinyeong@google.com
Google
Seoul, Republic of Korea

Juho Kim
juhokim@kaist.ac.kr
School of Computing, KAIST
Daejeon, Republic of Korea

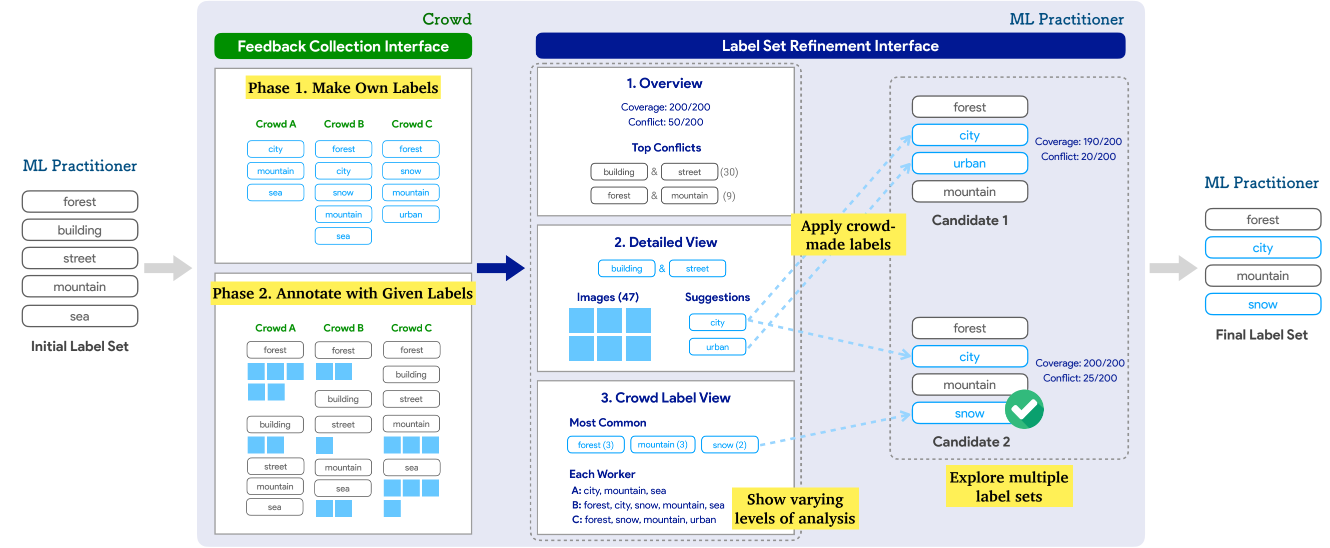


Figure 1: Label set refinement workflow using DynamicLabels. The initial label set of the ML practitioner is given to crowd workers with the *feedback collection interface* (green), where crowds could make their own labels (Phase 1) and annotate with ML-practitioner-made labels (Phase 2). The collected feedback is presented to the practitioner with the *label set refinement interface* (blue) through three varying levels of analyses, and the practitioner can apply crowd-made labels and explore multiple label sets to refine their label set.

ABSTRACT

Label set construction—deciding on a group of distinct labels—is an essential stage in building a supervised machine learning (ML) application, as a badly designed label set negatively affects subsequent stages, such as training dataset construction, model training, and model deployment. Despite its significance, it is challenging for ML practitioners to come up with a well-defined label set, especially when no external references are available. Through our formative study (n=8), we observed that even with the help of external references or domain experts, ML practitioners still need to go through multiple iterations to gradually improve the label set. In this process, there exist challenges in collecting helpful feedback and utilizing it to make optimal refinement decisions. To support informed refinement, we present DynamicLabels, a system that aims to support a more informed label set-building process with crowd feedback. Crowd workers provide annotations and label suggestions to the ML practitioner's label set, and the ML practitioner can review the feedback through multi-aspect analysis and refine the

[*] Jeongeon conducted this work during her Master's program at the School of Computing, KAIST.
[†] Jinyeong conducted this work while he was at NAVER CLOVA.

IUI '24, March 18–21, 2024, Greenville, SC, USA
© 2024 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-0508-3/24/03
https://doi.org/10.1145/3640543.3645157

label set with crowd-made labels. Through a within-subjects study (n=16) using two datasets, we found that DynamicLabels enables better understanding and exploration of the collected feedback and supports a more structured and flexible refinement process. The crowd feedback helped ML practitioners explore diverse perspectives, spot current weaknesses, and shop from crowd-generated labels. Metrics and label suggestions in DynamicLabels helped in obtaining a high-level overview of the feedback, gaining assurance, and spotting surfacing conflicts and edge cases that could have been overlooked.

CCS CONCEPTS

• **Human-centered computing → Interactive systems and tools**.

KEYWORDS

crowdsourcing, machine learning, label set construction, artifact or system

ACM Reference Format:
Jeongeon Park, Eun-Young Ko, Yeon Su Park, Jinyeong Yim, and Juho Kim. 2024. DynamicLabels: Supporting Informed Construction of Machine Learning Label Sets with Crowd Feedback. In *29th International Conference on Intelligent User Interfaces (IUI '24), March 18–21, 2024, Greenville, SC, USA.* ACM, New York, NY, USA, 20 pages. https://doi.org/10.1145/3640543.3645157

1 INTRODUCTION

A 'label' or a class, is a word or a phrase that explains a piece of data in a supervised machine learning (ML) model. A group of distinct labels works together as a 'label set' to provide the model with a set of candidate labels for classification [11]. For example, in classifying a clothing dataset, a label set may consist of four distinct labels: top , bottom , outer , and accessory . The label set is provided to the annotators to construct a training dataset and is utilized as model inputs and outputs. It is utilized often for classification tasks, such as the fashion classification model.

Preparing a well-constructed label set is important to build a successful ML application. Building an ML application involves a multi-stage process, which includes (1) preparing the raw data, (2) building a label set, (3) using the label set to annotate the training data, (4) implementing and training the model, and (5) deploying the model. Every other stage in the process is highly interconnected with the label set building stage: an unclearly defined label set affects the outcome of the annotation, and an indistinct or low-coverage label set affects the performance of the model, which subsequently negatively affects the experience of the user in the deployment stage [26].

When building label sets for classification models, ML practitioners usually refer to existing labeled datasets or theories (e.g., referring to existing psychology taxonomies for emotion recognition models) to come up with the label set [9], and iteratively validate and refine it with additional data [23]. The iterative refinement is essential for constructing a high-quality label set in many real-world situations. For example, applying a pre-established label set to real-world data requires revision of the label set to accurately represent the distribution of the data. In addition, building a label set from scratch for a domain without an established taxonomy requires a significant amount of feedback and consensus-building among ML practitioners. With multiple iterations, ML practitioners collect bad signals (e.g., low coverage, unclear distinction) on the label set and revise with the signals to prevent possible downstream issues, which is critical to the success of the ML application [23].

To further understand the practices and challenges of building label sets with iterative refinements, we conducted a formative study with eight ML practitioners who have experience constructing label sets from scratch. ML practitioners, even with existing references or domain experts, found revision cycles to refine and verify the label set to be important and necessary. During this process, they found it challenging to collect large-scale, fresh-perspective feedback to improve the label set. They also found it difficult to extract meaningful insights from the feedback and confidently decide on an optimal label set, due to many different aspects (e.g., clarity of each label, distribution of the data, clear boundary between the labels) they have to consider along with the uncertainty of each improvement decision.

To support collecting meaningful feedback and making informed decisions for refining the label set, we propose the idea of inviting crowd workers to provide feedback about the label set from varying perspectives, inspired by successful feedback mechanisms in the past to aid expert workflow [10, 21, 33], and leverage those feedback in designing interactions to better understand the data and make sufficient refinements. With the crowd as potential users of the deployed model, having the crowd's collective opinions and suggestions on the ML practitioner-built label set will guide the refinements. Providing analysis support through crowd feedback will help ML practitioners make a more confident and knowledgeable refinement to the label set.

To explore the proposed idea, we present DynamicLabels, a novel system that supports ML practitioners to iteratively construct their label set with label feedback collected from the crowd. When an ML practitioner provides an initial version label set, crowd workers produce feedback by annotating with the ML practitioner's label set and making their own label set with the assigned data through the *feedback collection interface*. With the collected label feedback and suggestions, the ML practitioner is provided with multiple-aspect feedback analyses and a playground to test and iterate on their label set in the *label set refinement interface*.

We conducted a 2-day within-subjects study comparing DynamicLabels with the baseline annotation system to examine how DynamicLabels supports an informed label set refinement with crowd feedback. A total of 16 ML practitioners used two types of datasets (natural scene images and event fliers) for a multi-class classification model to construct and refine two label sets through a round of iteration. The feedback collection interface of DynamicLabels enabled collecting large-scale, diverse feedback from the crowd, which participants identified as meaningful and useful in understanding the crowd's perspectives and the weaknesses in their label set to make refinements. The refinement interface of DynamicLabels enabled a high-level understanding of the feedback, encouraged flexible refinements to be made, and supported a structured refinement process. In addition, it helped the participants spot possible issues and examine various refinement options. We also discuss how DynamicLabels can support various types of data as well as the goals of ML practitioners. In addition, we suggest further utilization of the crowd feedback in making better-informed decisions, potential development to DynamicLabels in supporting automated and advanced support, and discuss how DynamicLabels supports the construction of a user-centered model.

Our contributions are as follows:

- DynamicLabels, a system that supports ML practitioners' label set construction process with crowd feedback and feedback analysis. DynamicLabels consists of a feedback collection interface that collects annotation and label suggestions on the ML practitioner-built label set, and a label set refinement interface that supports ML practitioners to make comprehensive refinement decisions.
- Findings from the formative interview with ML practitioners that examines the iterative verify-refine cycle of the label set construction process and the existing challenges.
- Findings from a within-subjects study that compares DynamicLabels with a baseline system—a crowd annotation system—using two datasets, which shows that DynamicLabels supports an exploratory and structured refinement process, and an in-depth analysis of how participants utilized crowd feedback in making label set refinements.

2 RELATED WORK

To situate our research, we first investigate existing approaches and challenges in the label set construction process. Then, as this work utilizes crowd feedback to support ML practitioners' label set building, we discuss how crowdsourcing is used to aid expert work. We also review decision-making supports enabled by large-scale data and visual analytics.

2.1 Label set construction for multi-class classification

When training a multi-class classification model, the ML practitioner should prepare an annotated dataset. Without external references, constructing a label set is more challenging as there is no standard practice in categorizing contents for a multi-class label set. One commonly used approach is applying clustering algorithms, such as LDA [1] and EM with GMM model [34]. These algorithms work in an unsupervised manner and categorize data points to compose clusters. However, these algorithms are mostly limited to numerically represented structured data. When the data contents are complicated and unstructured, other additional numerically abstracting algorithms or models are required to use these algorithms. Furthermore, they often fail at achieving reliable performance because these algorithms may not work perfectly. In addition, machine-generated clusters may not have appropriate representations or labels for human understanding.

To mitigate the issues from machine-generated label sets, previous work has invited humans to participate in the taxonomy or label set construction process [2, 7–9]. Cascade [9] presents a crowdsourcing workflow where workers provide suggestions and vote for the best descriptions over iterations to generate reliable categories with the crowds. Alloy [8] suggests a human-machine hybrid workflow to cluster text clips. A machine categorizes the text clips leveraging the salient keywords identified by crowd workers, then they put additional effort into clustering machine-failed clips. However, they rely entirely on the crowd and the machine, which can leave out practical considerations that an ML practitioner could make with intuition and experience. In addition, Revolt [7] leverages disagreement of crowds' annotation on a data instance to build label sets. They motivate that the labels and the descriptions created from disagreements can support an understanding of potential ambiguities in the label set. This creates more opportunities for ML practitioners to review and apply subjective labels, but is only investigated in binary classification scenarios.

In this work, we extend from prior works supporting the label set construction process with human work. DynamicLabels investigates the label set process after considering multi-stakeholders' (the crowd's and the ML practitioner's) perspective and for multi-label classification which is more complex. To our best knowledge, this is the first work that investigates label set construction from the ML practitioners' perspective, with crowds as feedback givers.

2.2 Collaborating with the crowd to support expert work

We define the label set construction as an open problem where no one best solution exists, so offering a diverse range of responses would help ML practitioners find an optimal label set satisfying their needs. Previous studies have shown that crowd inputs can help expert work as feedback [10, 21, 33] and inspiration for improvements [3, 15, 29].

Previous work leveraged crowd input as feedback to expert work. Voyant [33] collected structured crowd feedback on visual designs by providing five feedback types to the crowd. ProtoChat [10] collected multiple levels of feedback including utterance-level feedback and overall conversation feedback by asking questions while testing the conversation. CrowdCrit [21] introduced key sources in visual design for the crowd to refer to in making a critique, to collect detailed and actionable feedback.

Other work emphasizes the importance of incorporating aggregated crowd opinion in high-level concept or design of a product whose end user is a wide range of the public. Zhang et al. [37] asks the crowd to evaluate and cluster search results to present quality and satisfactory search results. Sutton and Lawson [29] proposed democratizing emoji design and selection by reflecting on how the public recognizes and uses emojis. Brambilla et al. [3] proposed a collaborative development process of Domain-Specific Modeling Languages in which end users and crowd workers are invited to provide feedback on diverse concepts of the language. In addition, Perspective [28] provided a set of auxiliary images and guiding questions to identify a diverse set of atypical images.

Our work is inspired by the approach of inviting the crowds as feedback provider to expert work, successfully investigated in various domains including designers ([10, 21, 29, 33]) and engineers ([3, 28]). DynamicLabels collects crowd annotations which can illustrate potential problems in the ML practitioner-built label set (e.g., confusion between labels or limited coverage of the label set). In addition, by asking crowds to design their own label set, DynamicLabels collects natural feedback with fresh perspectives on the label set. This allows ML practitioners to explore and examine various opportunities in early-stage.

2.3 Data-driven decision-making support

To make an informed decision with crowd feedback, ML practitioners need to understand the feedback thoroughly. Many previous work has explored ways to present data in a way that users can easily comprehend and utilize [13, 33, 36]. Voyant [33] automatically generates a word cloud with the collected feedback which is more helpful. Decipher [36] aggregates multiple feedback and provides a visualization tool to help the interpretation. Mudslide [13] helps teachers interpret the students' muddy points by visualizing students' feedback on lecture slides. Some studies have emphasized the importance of presenting a multi-faceted data analysis beyond aggregation [18, 32]. OpinionSeer [32] provides analysts with an interactive opinion visualization to easily explore the mined opinions. Kairam and Heer [18] used clustering techniques to leverage disagreement between crowd workers and showed that identified patterns could illustrate the worker characteristics as well as potential task problems.

Data-driven decision-making also enables users to consider various alternatives before making a decision. For conference session scheduling, Cobi [19] uses preference and constraints on metadata and presents the preview of changes in the number of conflicts for each move or assignment action that users consider. ConceptVector [24] supports an interactive construction of lexicon-based concepts by showing relevant documents and keywords regarding concepts the user considers. In designing a content-based image retrieval system for pathologists, Cai et al. [6] introduced tools that users can refine the image search by region, example, and concept.

Our label set refinement interface is inspired by previous work that supports a thorough understanding and consideration of various alternatives in decision-making, but is investigated in the unique context of label set construction. The label set refinement interface of DynamicLabels presents varying levels of crowd feedback, ranging from raw crowd annotation to estimated coverage and confusion, so that users can consider diverse aspects of the label set simultaneously to help users comprehend different aspects of the collected feedback. Also, users can preview the consequence of each change before making a refinement and construct and compare multiple versions of the label set.

2.4 Visual analytics for exploring and improving noisy data

In tasks such as ML model construction where a large volume of noisy data is utilized, experts often face difficulties in effectively understanding and improving the quality of the data if necessary. To support this process, many prior work has explored visual analytics as a plausible approach.

A line of work [5, 35] takes an automatic approach to correcting label errors and improving the performance of the classifier. Bäuerle et al. [5] categorizes three potential labeling errors and presents an automatic error detection approach to identify and resolve such errors. In addition, Yang et al. [35] proposes a visual analysis method, FSLDiagnotor, to automatically predict underlying causes for few-shot classifiers and improve them.

Another line of work involves human judgments in the process and supports effective exploration [17, 25, 30] and correction [20, 38] of necessary data. In supporting data exploration, Willett et al. [30] provide analysts with color clustering of crowd-generated explanations to quickly assess the collected data, and Park et al. [25] provides multiple views for crowdsourced medical annotation results to gain insights into the collected data. Moreover, Hoque et al. [17] utilizes a self-supervised learning approach to extract visual concepts to understand data at scale with minimal human effort. Other work takes an additional step to streamline the process by supporting the refinement of such data, where Liu et al. [20] utilizes three unique visualizations (confusion, instance, and worker) to assist experts in verifying uncertain instance labels and unreliable workers and LabelVizier [38] presents a human-in-the-loop workflow that assists in spotting and correcting incorrect annotations.

However, while the above-mentioned approaches concentrate on enabling a more accurate and efficient process, our work focuses on observing a complete end-to-end process of label set construction. DynamicLabels follows the approach of utilizing visual analytics to explore and make refinement decisions by supporting the exploration of crowd feedback with multiple-level analyses.

3 FORMATIVE STUDY

To understand the practice and challenges of ML practitioners in building and refining label sets for multi-class classification models, we conducted an hour-long interview with eight ML practitioners.

3.1 Procedure

The recruitment was done through various AI/ML communities online. The participants consisted of two ML research engineers, three ML engineers, two ML graduate researchers, and one AI planning product manager, and all had one or more experiences building label sets and datasets from scratch (Detailed descriptions in Table 1). The tasks they worked on were classification and object detection tasks, but the type of dataset varied from OCR tasks (involving business cards, receipts, and invoices), NLP tasks (involving online articles, chat messages, video transcript), to computer vision tasks (involving book covers, supermarket products, objects for autonomous driving). The size of the dataset they constructed ranged from thousands to ten-thousands, and the number of labels in the label set ranged from 10s to even 30s. The participants were compensated with KRW 50,000 (USD 40) for the interview.

The interview was conducted in a semi-structured format. We asked questions regarding their past label set construction experience, the aspects they consider important in building a label set, and the challenges and needs in constructing and refining label sets. For participants with multiple label set construction experiences, we additionally asked questions to compare and contrast the challenges and the experiences.

3.2 Practice

General process. All participants described label set building as one of the most challenging processes in constructing a training dataset, as the process involves coming up with an entirely new label set (i.e., coming up with a set of clearly described labels as well as detailed descriptions of each label) which involves numerous decisions. Even with relevant taxonomies or experts in the domain, the participants mentioned that *additional modifications are crucial* for the purpose of ML model construction such as granularity or

Table 1: Background information of formative study participants.

PID	Occupation	# of Label Set Building Experience	Primary Type of Dataset
P1	ML research engineer	6	Document data (business cards, legal documents)
P2	ML research engineer	2	Invoices
P3	AI planning product manager	3	Invoices, receipts
P4	ML engineer	5	Book covers, online articles
P5	ML engineer	3	Chat messages
P6	ML engineer	1	Grocery images
P7	ML graduate researcher	1	Video transcripts
P8	ML graduate researcher	1	Objects for autonomous driving

defining each label. For example, P7 (video context classification) referred to information classification taxonomy for initial label set construction, but had to largely modify the label set to fit to video context. P5 (psychological disabilities classification) worked with a professional psychologist but had to redefine the labels with more distinctive criteria considering the model. P8 described such modifications as "a decision area for the model builder to make."

The verify-refine cycle. The participants described their practice of iteratively refining the label set as going through *verify-refine cycles.* They first sample a small proportion of the data to construct an initial version of the label set. Then, they sample a larger amount of data and use the data to annotate with the label set. By looking at the annotations made, they decide whether the current label set is clear and appropriate to construct label sets and models. When the feedback from the verification shows issues with the label set (e.g., too many incorrect annotations, mixed use of labels), the participants make refinements. They mentioned that this cycle continues until no major issues are found in the label set, then proceeds to annotate to build the dataset. When there is more than one person involved in the construction process (e.g., as a team or with external annotators), they would compare and discuss the conflicts in the annotation.

Importance of a well-constructed label set. The participants also emphasized the importance of building a robust label set that can prevent latent issues, especially defining each label clearly and distinctly from each other while covering all edge cases. If not, wrong annotations can be made due to a misunderstanding of labels, potentially leading to biased dataset construction, poor quality of the model, and bad user experience with deployment. While some issues can be handled on the model side using existing techniques such as data augmentation, sometimes starting again from scratch is costly but inevitable. Considering the cost, they commented that they would rather iterate early in the label set construction (P2).

3.3 Challenges

Following the iterative nature of the label set construction, we identified three key challenges that ML practitioners face in the refinement process.

3.3.1 Lack of helpful feedback to improve the label set.

To improve their label set, it is common for ML practitioners to go through multiple feedback loops. The most prevalent way to collect feedback on the label set is by annotating using the constructed label set and spotting problematic data that cannot be covered or has conflicts. With the spotted data, the ML practitioner would make refinements to the label set until they are convinced that all major issues are resolved. While this approach helps in collecting problematic data and making refinements by adding labels or new descriptions for edge cases, the participants mentioned the limited help that annotations can provide. When the annotation is done by the ML practitioners themselves, it is more difficult to spot uncovered or conflicting data due to biases towards certain labels. P4 mentioned that "even if they go through multiple iterations within the team, there are always unexpected questions asked by the annotators." To get fresh feedback on the label set, ML practitioners sometimes recruit external annotators. This is more effective than having the team annotate as the problematic data are collected based on the annotator's perspective. However, often the ML practitioners "end up adding a bunch of rule-based descriptions" (P2), which results in inefficiency and confusion for the annotators in making the training dataset later. In addition, recruiting external annotators to perform annotation can be a troublesome and costly process (P7).

3.3.2 Difficulty in comprehending meaningful insights from the feedback.

As mentioned previously, constructing an optimal label set is difficult due to the many aspects (e.g., clear distinction of the labels, clear description of the labels) and multiple stakeholders (e.g., annotators, the requester of the model, users of the model) that need to be considered together. While each ML practitioner has a set of criteria they consider important, there is no clear guideline on making an optimal label set, making it difficult for them to decide on the best label set. Thus, they rely heavily on feedback in the construction process.

This complex nature of label set construction makes extracting high-criticality insights from feedback challenging. When feedback about a label set is collected in the form of issues or annotation results, ML practitioners need to examine each piece of feedback and organize them to come up with a concrete revision item. However, it is challenging for them to both find critical feedback from a bunch of collected feedback and group them into a meaningful revision item. More specifically, P2 mentioned that "edge cases that lead to adding a description is relatively easy, while [those] that lead to a change in hierarchical structure and definition is very difficult to spot and decide." For conflicting labels, the participants mentioned that when annotators use a different label than their original intention, they know that something is confusing. However, it is difficult to decide whether the situation is common and should be prioritized.

3.3.3 Difficulty in utilizing the insights to make satisfactory changes. After ML practitioners organize key insights from the feedback, they need to apply the insights to modify the label set. Understanding the insights does not mean that a suitable modification can be made to the label set, as a complex set of criteria must be considered. As a result, ML practitioners are often not sure about their changes and their consequences.

ML practitioners try other approaches to increase certainty in the decision-making process such as discussing with a team of ML practitioners. However, this can be time-consuming and difficult to reach a consensus. Even when a consensus is reached, there is no guarantee that the decision is optimal. The only way to know whether the decision is optimal or not is by getting to the later process of the ML model construction process (e.g., dataset construction, model training) and seeing if any issues occur. Without being able to check, participants struggled and felt less confident in choosing the right moment to proceed to dataset construction, and stated "I'm afraid that the label set will end up creating issues later in the model building process" (P3).

3.4 Design Goals

Based on the interview results, we came up with the following design goals for a system that addresses the challenges ML practitioners face in iteratively building and refining label sets.

3.4.1 Collect helpful feedback on the label set from the crowd. ML practitioners identified a need for collecting nutritious feedback in their label set construction process to find problems in the label set and make appropriate refinements. Specifically, the participants mentioned the need to receive feedback from fresh perspectives, that more actively suggest possible changes, and on a larger scale to address as many issues as possible. Through crowdsourcing, a group of people having fresh, diverse perspectives can be recruited to collect large-scale feedback on the label set (C1). In addition, the crowd can also provide their own labels as suggestions to support the refinement process as well (C3).

3.4.2 Provide multi-aspect analysis to derive meaningful insight. One major characteristic of label set construction is that there is no best practice for an optimal label set. Thus, ML practitioners face difficulty interpreting and obtaining meaningful insight from the feedback they collect. During the interview, participants stated that they mainly get a sense of problematic labels by examining the feedback in varying aspects. Likewise, showing the collected feedback in multiple aspects (e.g., highlighting conflict, showing edge cases, providing a summary) can support ML practitioners to thoroughly understand the feedback and select ones to prioritize.

3.4.3 Help understand possible changes and consequences in the label set. Even after extracting meaningful insights from the feedback, ML practitioners struggle to make confident changes to the label set due to the uncertainty of their action consequence. Actively supporting ML practitioners with possible label candidates or showing them the consequence of the label set with the changes will help the refinement process be more informed, and will lower the barrier to iterate on the label set.

4 PROPOSED SYSTEM: DYNAMICLABELS

We present the design of DynamicLabels, a system that aims to support ML practitioners' label set construction. DynamicLabels supports iterative refinement of the label set through two separate interfaces: the **feedback collection interface** and the **label set refinement interface**. The former is provided to the crowd to collect annotations and label suggestions on the ML practitioner-built label set, and the latter is provided to the ML practitioners for refinement with multiple analyses of crowd feedback.

Overall label set construction workflow and the role of each interface in DynamicLabels are described in Figure 1. In DynamicLabels, label sets are constructed in tree form (As in Appendix A.3 Figure 11), consisting of labels and groups to group the labels. Each label includes a label name and a description [1]. As for the scope of research, we allow a single label to be assigned to each image to simulate the simplest form of label set.

4.1 Feedback collection interface

For the crowd workers to use the feedback collection interface, the practitioner needs to have a constructed label set beforehand. This is similar to the practice in real-life settings, where the practitioners first build an initial label set.

The crowd is asked to provide feedback through two phases: (1) providing label suggestions by making the crowd's own label set and (2) annotating with the ML practitioner-built label set (Fig. 1-Feedback Collection Interface). Two types of feedback are collected: passive (annotation results) and active (new label creation as suggestions), where an active suggestion aims to collect diverse perspectives on the label set. [2]

4.1.1 Phase 1 - Providing label suggestions by making the crowd's own label set. Crowd workers start from the Phase 1 task: creating their own label set (Fig. 2). They are first asked to take a look at 30 assigned images (Fig. 2-b) and come up with a set of labels (Fig. 2-a). Then, they are instructed to use the labels to make annotations (Fig. 2-c).

4.1.2 Phase 2 - Annotating with the ML practitioner-built label set. Crowd workers then proceed to the next phase and use the ML practitioner-built label set to annotate the same 30 images (Fig. 3). In addition to the ML practitioner-built label set (Fig. 3-a), the workers are provided with an additional others label to annotate images that do not fit into the provided label set to spot edge cases. For each image labeled others, the workers are asked to provide a brief reason (Fig. 3-d) to justify their choice.

4.1.3 Post-processing of crowd feedback. Crowd annotations and crowd-made labels are post-processed to provide meaningful analyses. To avoid suggesting redundant crowd-made labels, we merged

[1] Following the label format from Google documentation (https://cloud.google.com/ai-platform/data-labeling/docs/label-sets)

[2] To prevent biases, the crowd is asked to build their own label set before annotating with the ML practitioner-built label set.

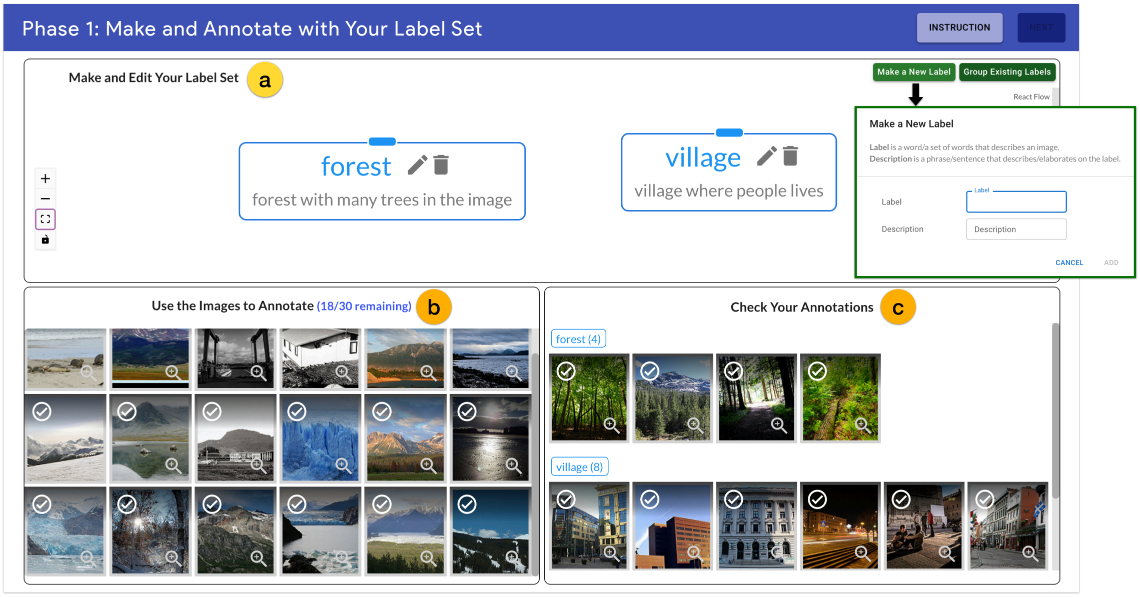

Phase 1: Make and Annotate with Your Label Set
INSTRUCTION
NEXT
Make and Edit Your Label Set
a
Make a New Label
Group Existing Labels
React Flow
forest
forest with many trees in the image
village
village where people lives
Make a New Label
Label is a word/a set of words that describes an image.
Description is a phrase/sentence that describes/elaborates on the label.
Label
Description
CANCEL
ADD
Use the Images to Annotate (18/30 remaining)
b
Check Your Annotations
c
forest (4)
village (8)

Figure 2: Phase 1 of the feedback collection interface. The crowd workers are instructed to check the assigned images through (b) a grid of images on the bottom left and make their own label set on the (a) top component by adding, revising, and deleting the labels. For created labels, they can select the images in (b) to annotate, which will show up in (c), under each label.

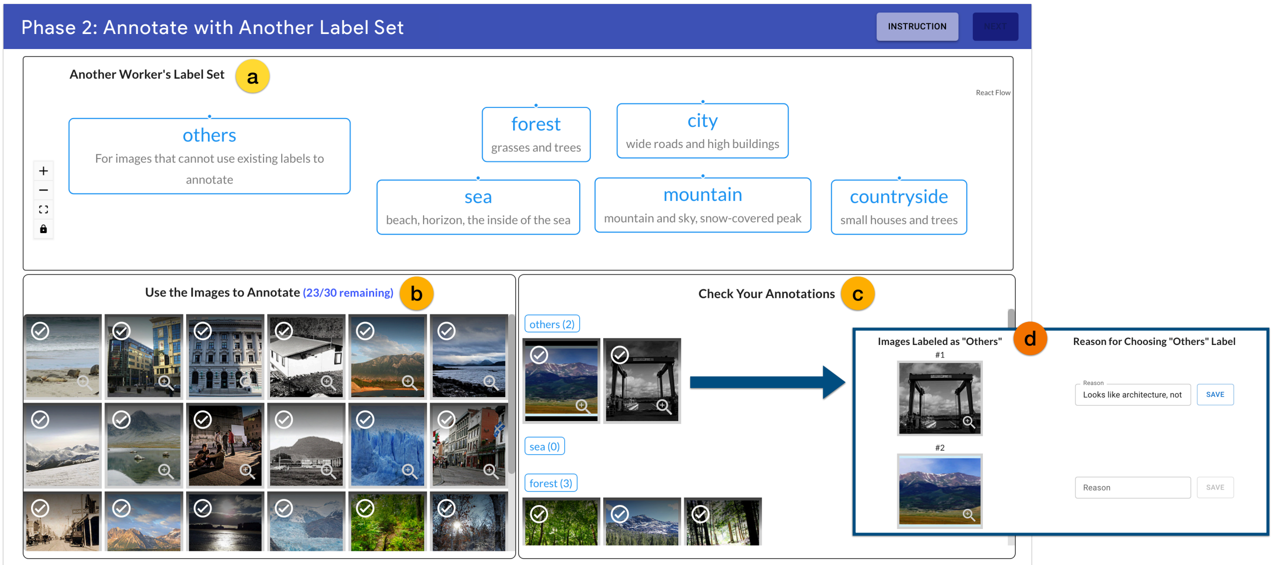
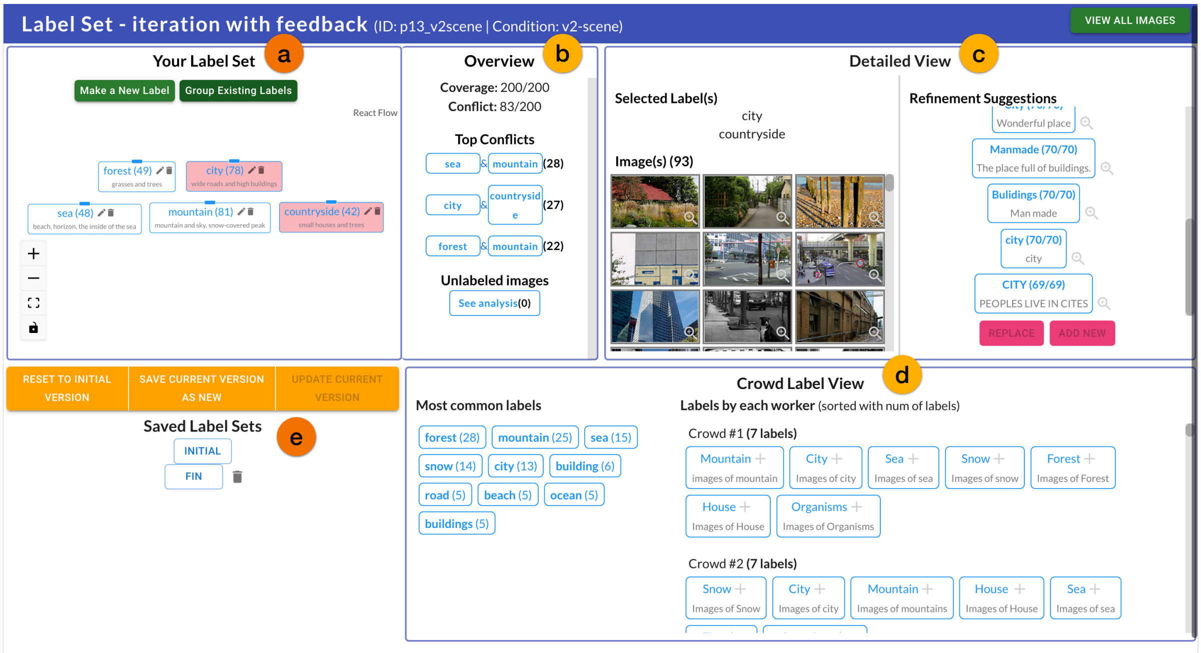

Phase 2: Annotate with Another Label Set
INSTRUCTION
NEXT
Another Worker's Label Set
a
React Flow
others
For images that cannot use existing labels to annotate
forest
grasses and trees
city
wide roads and high buildings
sea
beach, horizon, the inside of the sea
mountain
mountain and sky, snow-covered peak
countryside
small houses and trees
Use the Images to Annotate (23/30 remaining)
b
Check Your Annotations
c
others (2)
sea (0)
forest (3)
Images Labeled as "Others"
d
Reason for Choosing "Others" Label
#1
#2
Reason
Looks like architecture, not
SAVE
Reason
SAVE

Figure 3: Phase 2 of the feedback collection interface. The crowd workers are instructed to take a look at the (a) ML practitioner’s label set and use the labels to annotate the (b) assigned images. Annotations will show up in (c), under each label. For images annotated using the “others” label, the workers are asked to provide a (d) brief reason each as an additional step.

multiple crowd-made labels into one if they are identical after stemming and lemmatizing.

As each crowd worker makes their own labels based on 30 images, the number of crowd annotations, or the number of images, for each crowd-made label is limited to 30 at most. To help ML practitioners better estimate the coverage and potential confusion of crowd-made labels, we established extended annotation for crowd-made labels. We first established similarity relationships between crowd-made labels and ML practitioner-made labels. For each crowd worker, we calculated the Jaccard similarity coefficient for each pair of a crowd-made label and an ML practitioner-made label, based on the crowd worker’s annotation of 30 images for ML practitioner-made



Label Set - iteration with feedback (ID: p13_v2scene | Condition: v2-scene)

Coverage: 200/200
Conflict: 83/200

Top Conflicts: sea & mountain (28); city & countryside (27); forest & mountain (22)

Unlabeled images: See analysis(0)

Selected Label(s): city, countryside

Image(s) (93)

Refinement Suggestions: Wonderful place; Manmade (70/70) The place full of buildings.; Bulidings (70/70) Man made; city (70/70) city; CITY (69/69) PEOPLES LIVE IN CITES

Most common labels: forest (28), mountain (25), sea (15), snow (14), city (13), building (6), road (5), beach (5), ocean (5), buildings (5)

Labels by each worker (sorted with num of labels)

Crowd #1 (7 labels): Mountain – images of mountain; City – Images of city; Sea – Images of sea; Snow – Images of snow; Forest – Images of Forest; House – Images of House; Organisms – Images of Organisms

Crowd #2 (7 labels): Snow – Images of Snow; City – Images of city; Mountain – Images of mountains; House – Images of House; Sea – Images of sea

Figure 4: Overview of the label set refinement interface. The (a) current version label set is displayed at the top left, along with an (b) overview of the collected feedback. By clicking the labels in (a) or top conflicts and unlabeled images in (b), the ML practitioner can see a (c) detailed view. On the bottom right, you can see the (d) crowd label view. During the refinement, you can save different versions of the label sets, which are displayed through the (e) saved label sets.

labels and crowd-made labels. For pairs with a similarity higher than 0.8, we assumed that the crowd label and the ML practitioner-made label are similar. Then, for each ML practitioner-made label, we filtered out images whose majority vote (of crowd annotation) match the label and established extended annotation between those images and crowd-made labels with high similarity with the ML practitioner-made label.

4.2 Label set refinement interface

When a sufficient amount of feedback is collected for each image, the ML practitioner can revise their label set through the label set refinement interface (Fig. 4) with the following components: Your label set, Overview, Detailed view, Crowd label view, and Saved label sets. The interface supports reviewing and understanding the feedback with three different analyses (Fig. 4-b,c,d), and adopting the feedback to make changes with crowd-made labels (Fig. 5).

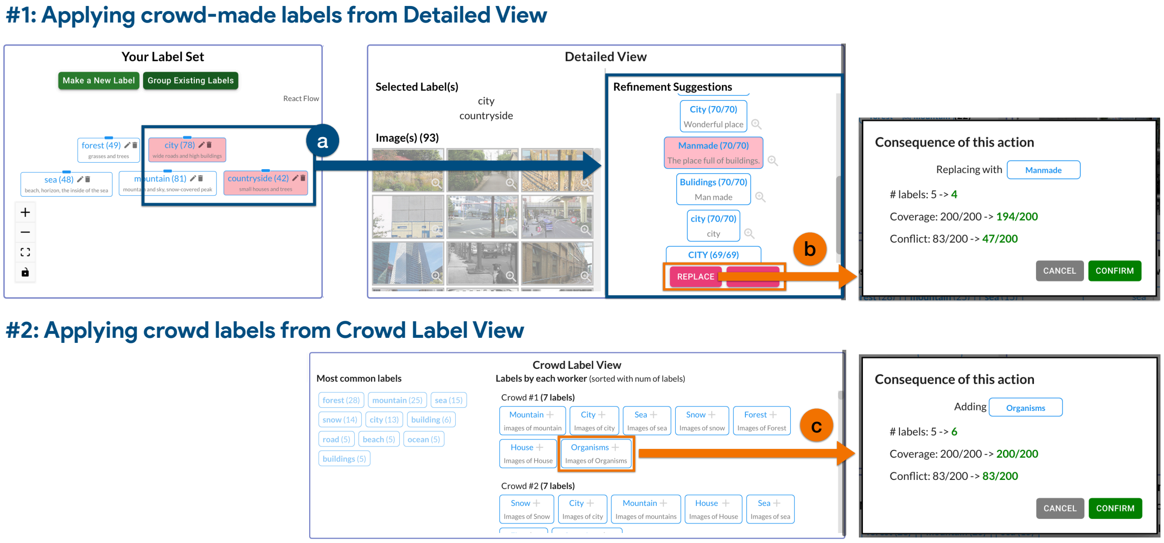
4.2.1 Showing varying levels of analysis for the collected feedback. When the ML practitioner enters the label set refinement interface, they can find their initial label set on the top left, under "Your label set". Right next to it is an **overview** (Fig. 4-b) that shows a summary created with the crowd feedback. Inside the overview, we provide four metrics motivated by the formative study, (1) Coverage: number of images with annotation, (2) Conflict: number of images annotated with multiple labels, (3) Top conflicts: top 3 label pairs with the highest number of conflicts, and (4) Unlabeled images: number of images without annotation to spot the main issue in their label set. The metrics are re-calculated when any changes are made to the label set. These metrics help ML practitioners understand how each label would be perceived and understood by the crowd and the coverage of labels as a set.

For the ML practitioner to understand the collected annotation in detail, they can select label(s), top conflicts, or unlabeled images to see a **detailed view** (Fig. 4-c). Here, the ML practitioner can see images annotated with the selected label(s) (current label set), images annotated using the conflicting labels (top conflicts), or images that are not labeled (unlabeled images) on the left. On the right, they can see possible refinement suggestions—a list of crowd-made labels that overlap the most with the selected set of images.

On the bottom right, the ML practitioner can explore refinement options, through an analysis of the crowd-made labels through the **crowd label view** (Fig. 4-d). DynamicLabels shows crowd-made labels in two different aspects: (1) Most common labels and (2) Labels by each worker. The Most common labels component shows the top 10 frequently-made crowd labels, and the Labels by each worker component shows a list of crowd-made label sets, sorted by the number of labels made.

4.2.2 Providing refinement support with crowd-made labels. To support a more informed refinement, we allow ML practitioners to make additions or replacements to their label set using crowd-made labels. The refinement actions can take place from refinement suggestions or labels by each worker, illustrated in Figure 5.

On each refinement action, we display the **action consequence modal** (right of Fig. 5), where the change in the overview (the number of labels, coverage, conflict) is shown before making the



Consequence of this action

Replacing with Manmade

labels: 5 -> 4

Coverage: 200/200 -> 194/200

Conflict: 83/200 -> 47/200

Consequence of this action

Adding Organisms

labels: 5 -> 6

Coverage: 200/200 -> 200/200

Conflict: 83/200 -> 83/200

Figure 5: Two possible ways to apply crowd-made labels to the current label set. In the top example, the ML practitioner can select (a) two labels in the current label set (city and countryside) to see a detailed view. From the refinement suggestions in the detailed view, the ML practitioner can (b) select crowd-made labels (Manmade) and click on the action (*replace*) to trigger the action consequence modal and make refinement decisions. In the bottom example, the ML practitioner can (c) click the plus icon next to the crowd-made label (Organisms) to add the label, which will trigger the action consequence modal.

change. The ML practitioner can use the model to decide whether to apply the refinement or not. In addition to the detailed view, the ML practitioner can add crowd-made labels individually through the Crowd label view component (Fig. 4-d). The same action consequence modal is shown for this refinement as well. The ML practitioner can also directly add new labels, edit existing labels, or delete existing labels (Fig. 4-a), while the overview metrics stay the same.

4.2.3 Creating and exploring multiple label set candidates. On the bottom left, there is a **saved label sets** component (Fig. 4-e), which enables exploration of potential candidates with version control.

5 STUDY DESIGN

We conducted a 2-day study with 16 ML practitioners to investigate how DynamicLabels assists the ML label set construction process, through a within-subjects study comparing DynamicLabels to the baseline system (Described in Section 5.2.2). We chose this study design because of high variance in ML practitioners' machine learning expertise, previous label set/dataset construction experience, and perception of crowdsourcing/crowd workers.

We aimed to answer the following research questions:

(1) Can crowd workers produce helpful feedback with the feedback collection interface?
(2) How do ML practitioners use crowd feedback to refine their label sets?
(3) How do ML practitioners use the refinement interface in DynamicLabels to make informed refinement decisions?

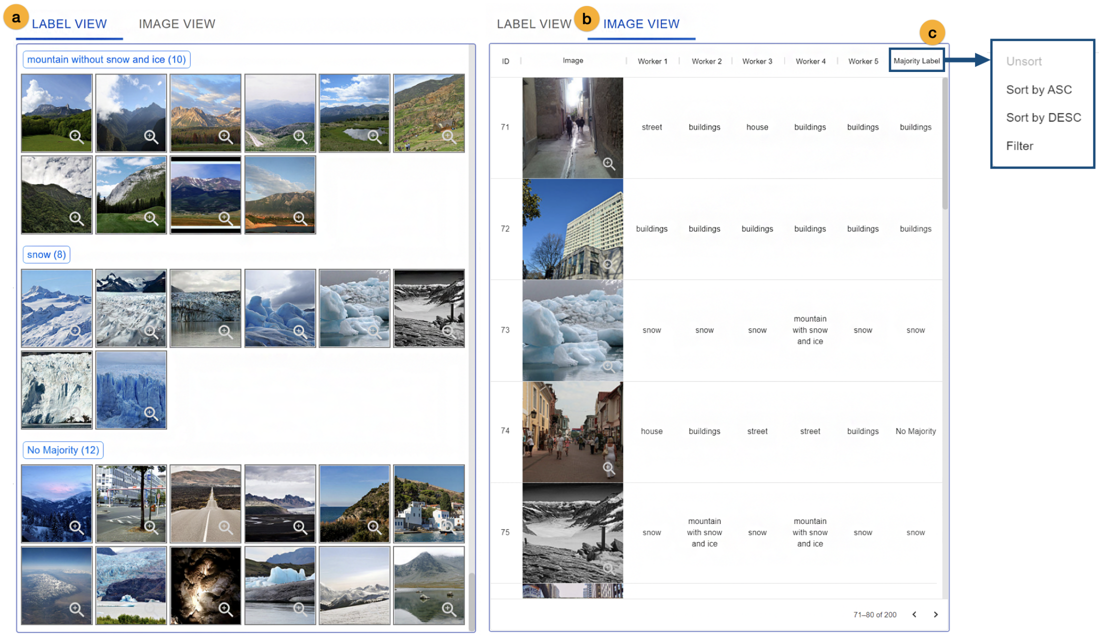
For the first RQ, we compare DynamicLabels of the collected feedback with that of the baseline system. For the third RQ, we compare and additionally explore how ML practitioners utilized the *label set refinement interface* through our suggested system. For the second RQ, we do not compare DynamicLabels with baseline but derive common patterns of ML practitioners utilizing crowd feedback in both conditions.

5.1 Participants

We recruited 16 participants by making an open call in several universities' online communities and social media targeting ML practitioners. Participation was limited to those with experience (1) manually constructing or utilizing label sets for ML models and (2) conducting industry or research projects using multi-class classification models that require label sets with multiple labels. Among 16 participants, 3 were undergraduate students, 7 were graduate students, and 6 were industry workers. All participants had experience making label sets for classification models. We describe detailed demographics and the task they used for the study in Appendix A.1. Each participant was compensated KRW 120,000 (USD 94) for a total of 4.5-hour 2-day participation.

5.2 Study Setup

5.2.1 Task and Datasets. For the study, we asked participants to design a label set for a multi-class classification model. The participants each defined a specific task they would use the model for and were asked to improve their label set through a single refinement cycle. We selected two types of data: natural scene image dataset [39] and event flier dataset (manually collected by the authors). We refer to the natural scene image dataset as *scene* and the event flier dataset as *flier* from below. We chose datasets that



ID	Image	Worker 1	Worker 2	Worker 3	Worker 4	Worker 5	Majority Label
71		street	buildings	house	buildings	buildings	buildings
72		buildings	buildings	buildings	buildings	buildings	buildings
73		snow	snow	snow	mountain with snow and ice	snow	snow
74		house	buildings	street	street	buildings	No Majority
75		snow	mountain with snow and ice	snow	mountain with snow and ice	snow	snow

Figure 6: How the feedback was provided to the practitioners in the refinement interface of the baseline system. On the (a) label view, the user can see each label with images whose majority winner is the label and those without majority winners. On the (b) image view, the user can see raw annotations and majority voting results for each image. The (c) majority label column on the image view can be clicked to sort or filter results.

do not require a high level of domain expertise for the crowd to understand while having a varied modality, *scene* having images only, and *flier* having images and texts.

From each dataset, we randomly sampled 200 images for the study. Among the 200, we randomly selected 50 images for the initial label set construction, and all 200 images for collecting crowd feedback and the refinement stage. We decided on the two numbers (50 and 200) based on the formative practice, where practitioners normally conduct the first iteration with hundreds of data.

5.2.2 Baseline System. Our baseline system (Figure 6) is designed with reference to the verify-refine feedback loop described during the formative study, where (1) crowd workers annotate each image to one of the ML practitioner-designed labels, and (2) the raw annotations and majority voting results are presented in the refinement phase (Figure 6).

5.2.3 Procedure. The study was conducted with two sessions to simulate a single iteration of label set construction (Detailed procedure in Figure 7). Each participant conducted the task in the baseline condition for one dataset and DynamicLabels condition for the other dataset. The order and the image types assigned to the conditions were counterbalanced and randomly assigned.

In the first session, the participants first created an initial version of the label set with 50 images. Constructed label sets were used to collect crowd feedback (DynamicLabels) or crowd annotation (baseline) for 200 images. After each label set construction, we asked the participants to fill out a 7-point Likert scale survey (1 = Strongly disagree, 7 = Strongly agree) regarding the construction process. After the participants finished constructing the two initial version label sets, we conducted a semi-structured interview regarding the construction process, the challenges in the process, and the participants' expectations of crowd feedback.

In between sessions 1 and 2, we collected crowd feedback (DynamicLabels) and annotations (baseline) on the label sets that participants constructed through Amazon Mechanical Turk [3]. For each label set, we recruited 34 crowd workers, and each image was annotated by five different workers. Each worker was assigned 30 images with a 24-image overlap with the previous worker. The workers were paid $8.0 per hour for their work. We limited participation to U.S. workers who had completed at least 1000 HITs with an approval rate of at least 97%.

In the second session, the participants were instructed first to look at the raw data of the collected feedback/annotation and rate the data on a 7-point Likert scale to evaluate the helpfulness of the collected data, dimensions including good quality, large-scale, diverse opinions/perspectives, usefulness/meaningfulness of the data. The participants then refined their label set with the crowd feedback (DynamicLabels) or the crowd annotations (baseline) collected. After each refinement task, we asked the participants to fill out a 7-point Likert scale regarding the refinement process, dimensions including whether the process was structured, whether

[3] https://www.mturk.com

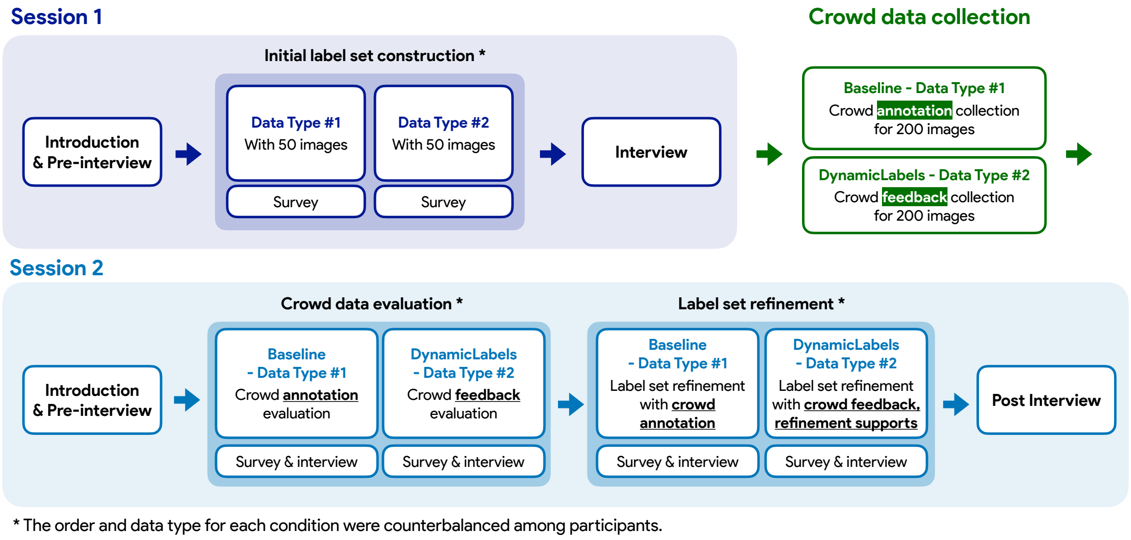

Session 1
Initial label set construction *
Introduction & Pre-interview
Data Type #1
With 50 images
Data Type #2
With 50 images
Survey
Survey
Interview
Crowd data collection
Baseline - Data Type #1
Crowd annotation collection for 200 images
DynamicLabels - Data Type #2
Crowd feedback collection for 200 images
Session 2
Crowd data evaluation *
Label set refinement *
Introduction & Pre-interview
Baseline - Data Type #1
Crowd annotation evaluation
DynamicLabels - Data Type #2
Crowd feedback evaluation
Baseline - Data Type #1
Label set refinement with crowd annotation
DynamicLabels - Data Type #2
Label set refinement with crowd feedback, refinement supports
Survey & interview
Survey & interview
Survey & interview
Survey & interview
Post Interview
* The order and data type for each condition were counterbalanced among participants.

Figure 7: Tasks and procedure for each session. In session 1, each participant creates two initial label sets for each dataset. The label sets are given to the crowd workers to collect annotation or feedback depending on the condition. In session 2, the participant refines their label set with the collected crowd data presented.

uncertainties were considered, whether alternatives were considered, and whether confident decisions were made (modifying Quality Decision-Making Procedure (QDMP) [4]). Afterwards, we conducted a semi-structured interview with the participant regarding the overall refinement process, utilization of collected crowd feedback/annotation, utilization of features in the refinement interface, and the refinement process and the final label set. After completing the two refinement tasks, we asked questions comparing the refinement process using DynamicLabels and the baseline system.

5.3 Measures

We collected and analyzed the following data: crowd feedback, participants' session 1&2 label sets and refinement logs, task observations and interview responses, and survey results on crowd feedback/annotation and label set construction/refinement process.

Crowd annotations. We measure the collected crowd feedback in terms of diversity, helpfulness, and quality. We use the total and unique number of crowd-made labels, and the unique number of labels used to annotate a single image to measure the diversity, and survey and interview responses on crowd feedback/annotation to measure the helpfulness.

As a quality measure, we measured the accuracy of crowd annotations. We first filtered out 52 out of 1,073 workers who showed clear trolling behavior. [4] Then, we randomly sampled 2,000 crowd annotations for participant-made labels (500 for each dataset and condition) and 1,000 crowd annotations with crowd-made labels (for DynamicLabels condition, 500 for each dataset) among 47973 annotations in total. With 3,000 sampled annotations, two of the authors coded the accuracy. As the collected labels were subjective, we considered annotations that appropriately describe or represent each image as correct (e.g., considering both buildings and city correct for the left image in Figure 9).

Refinement actions. We extracted each participant's refinement actions from session logs and recordings. Then we categorized each refinement action into seven categories: three label changes (add, revise, delete), one description-level change, and three group-level changes (add, revise, delete).

There were cases where multiple refinement actions were made to achieve one high-level refinement (i.e. split and merge). To analyze such high-level refinement actions made by participants, we grouped refinement actions made for one high-level split and merge refinement. Three of the authors analyzed three (out of 32) sessions together and then analyzed the remaining sessions individually.

We also recorded whether each refinement action was made based on crowd-made labels. In addition to direct use (adding or replacing with) of crowd-made labels, we also recorded actions where participants adopted crowd-made labels or descriptions.

Interview responses and session observations. Participants' think-aloud and interview responses were transcribed and analyzed through an open coding process, followed by focused coding. Two authors first individually developed a set of codes. Then, the two authors collapsed the developed codes into themes by identifying similar codes under each research question.

6 RESULTS

We first present an overview of the study results and then discuss each RQ in detail. The overview presents descriptive statistics of the label sets that participants made in sessions 1 and 2, refinements made in session 2, and the crowd labels and annotations collected for each participant's session.

[4] Those who used two or fewer ML practitioner-made labels to annotate 30 images or made out-of-context labels (e.g., making jacket or example1 in *flier*) were excluded.

6.1 Descriptive Statistics

6.1.1 Label set construction and refinements. In session 1, the participants constructed an initial label set with 50 images. The median number of labels was 7 (min: 3, max: 11) for *scene* and 9.5 (min: 4, max: 14) for *flier*. The initial label set construction on average took 21.4 minutes (σ=6.8) for *scene* and 41.9 minutes (σ=16.4) for *flier*.

In session 2, the participants refined their label set using the crowd annotations (baseline) or feedback (DynamicLabels) collected with 200 images. Table 2 shows the median number of labels and groups in the initial and revised label sets, and the number of net changes made in the labels and groups for each condition and dataset. Participants generated more labels with *flier* dataset (median: 9.5 with min: 4, max: 14) than with *scene* dataset (median: 7 with min: 3, max: 11). However, within each dataset, there was no statistically significant difference in the number of labels, groups, and changes between conditions.

Participants spent significantly more time refining the label sets with DynamicLabels than baseline and with *event* than with *scene* (two-way repeated ANOVA, F=8.38 with p<0.05 between conditions and F=4.84 with p<0.05 between datasets). With *scene* dataset, participants spent 16.8 (σ=10.6) minutes for the baseline and 19.5 (σ=11.4) minutes for DynamicLabels to refine their label sets. For the *flier* dataset, the average time spent was 17.4 (σ=13.1) minutes for the baseline and 32.4 (σ=16.3) minutes for DynamicLabels.

Table 3 shows the median number of refinement actions made in each condition for each dataset. With the baseline, a median of 8.5 (min: 2, max: 24) and 9.5 (min: 2, max: 23) refinement actions were made by participants *scene* and *flier* datasets, respectively. With DynamicLabels, participants made a median of 7 (min: 0, max: 25) and 11 (min: 5, max: 15) refinement actions for *scene* and *flier*, respectively. The specific refinement actions made by each participant under each condition are shown in Figure 10. Figure 11 shows an illustrative example of label set refinement made with DynamicLabels.

6.1.2 Crowd feedback. Between sessions 1 and 2, the crowd made annotations (baseline) or feedback—annotations and label suggestions (DynamicLabels)—using the feedback collection interface. Table 4 summarizes the crowd feedback collected for a single label set. For each ML practitioner-made label set, crowd workers made an average of 1053.63 and 1023.75 annotations for *scene* and *flier*, and 969.25 and 1023.75 annotations for *scene* and *flier* in the baseline system and DynamicLabels, respectively. In addition, for DynamicLabels the workers made 179.75 labels (72.13 unique labels) for *scene* and 199.25 labels (106.00 unique labels) for *flier* on average. The accuracy of annotations with ML practitioner-made labels was 88.59% (*scene*), 69.18% (*flier*) with the baseline, and 90.52% (*scene*), 72.88% (*flier*) with DynamicLabels. The accuracy of annotations with crowd-made labels was 91.60% in *scene* and 73.40% in *flier*.

For the baseline task, the average time spent was 827.57 seconds (σ = 1044.30) in *scene* and 1770.96 seconds (σ = 1484.30) in *flier*. For the DynamicLabels task, the average time spent was 1269.08 seconds (σ = 1401.17) in *scene* and 2181.93 seconds (σ = 1679.29) in *flier*. While the phase 1 task in DynamicLabels can be more mentally demanding than phase 2, requiring workers to create new labels, the time spent is similar to or less than twice the time for that of baseline. We presume that this was because the workers utilized the same set of images in phases 1 and 2, decreasing the time needed to understand and become familiar with the data in phase 2.

6.2 RQ1: Can the crowd produce helpful feedback with the feedback collection interface?

Feedback from the crowd—both annotation and crowd-made labels—contained diverse viewpoints and helped participants understand the various viewpoints of the crowd, some they had never expected before, and found it meaningful and useful in making refinements.

6.2.1 Diversity of Crowd-made Labels. A single crowd worker on average created 5.58 labels for *scene* and 6.18 labels for *flier*, summing up to on average 179.97 labels (*scene*) and 199.25 labels (*flier*) created per ML-practitioner's label set. When we counted the number of unique labels after lemmatization, the number of unique labels was 72.13 labels for *scene*, and 106.00 labels for *flier*. Considering that the average final number of labels that participants made was around 7 to 10, around 10x labels were provided to the participants per label set.

Crowd workers' diverse viewpoints were captured with the number of labels they created/utilized for a single image (Figure 8, examples in Figure 9). For crowd-made labels, on average 3.77 labels (*SD*=0.43, *event*) and 4.41 labels (*SD*=0.47, *flier*) were used to annotate a single image, whereas for ML practitioner-built labels, on average 2.07 labels (*SD*=0.39, *event*) and 3.32 labels (*SD*=0.46, *flier*) were used. The difference between the number of crowd-made labels and ML practitioner-built labels used was significant for both data types ($t(398)=40.95$, $p < .0001$ for *scene*, $t(398)=23.83$, $p < .0001$ for *flier*), exhibiting a wider perspective of the crowd in making their own labels, an aspect of helpful feedback mentioned in the formative study.

In follow-up interviews regarding the crowd feedback, participants noted that the crowd-made labels helped them make various interpretations of the crowd workers in perceiving and recognizing the datasets. P15 commented that "through the crowd-made labels, I can see how people perceive and categorize the datasets, which I cannot understand through looking at the consequent annotations."

6.2.2 Perceived Helpfulness. All participants commented that the crowd feedback was meaningful and useful in making refinements to the label sets. P14 said, "I was able to see a difference in my understanding and the crowd's understanding of the label, [...] the annotation is different from what I expected. However, this will be helpful [to me in the refinement] as this tells me that my label is poorly defined." Participants stated that seeing the crowd-made labels for each image in DynamicLabels makes up for the difficult-to-understand annotations, functioning as reasons. For example, P12 was confused about why a running flier was not being annotated as activity rather than sports, but understood the reason by seeing exercise and yoga labels made by the workers.

Participants especially liked the varying granularity of the crowd-made labels in DynamicLabels. P1 perceived the workers' label sets as "an evolution of the label sets from the most general to the most specific." They browsed through the crowd-made labels multiple times for opportunities to bring them to their own label sets.

Table 2: Median number of labels and groups made in Sessions 1 and 2 and the median number of net changes in the label set, for each dataset and condition. Participants created label sets with more labels and a higher range in *flier* compared to *scene*, while there was no difference in the labels, groups, and net changes between DynamicLabels and baseline.

			Natural Scene					Event flier				
			Session 1	Changes (net)			Session 2	Session 1	Changes (net)			Session 2
				Added	Deleted	Revised			Added	Deleted	Revised	
Baseline	# of labels	Median	6	1.5	1	2.5	7	9	1.5	1.5	2	8.5
		[Min, Max]	[3, 10]	[0, 6]	[0, 7]	[0, 5]	[3, 10]	[5, 16]	[0, 8]	[0, 9]	[1, 6]	[6, 13]
	# of groups	Median	1	0	0	0	1	0	0	0	0	0
		[Min, Max]	[0, 6]	[0, 1]	[0, 1]	[0, 1]	[0, 6]	[0, 4]	[0, 3]	[0, 2]	[0, 2]	[0, 3]
DynamicLabels	# of labels	Median	7	2	3	0.5	7	9	2	1.5	1.5	10
		[Min, Max]	[5, 11]	[0, 4]	[0, 1]	[0, 4]	[5, 10]	[4, 14]	[0, 6]	[0, 4]	[0, 10]	[6, 15]
	# of groups	Median	0.5	0	0	0	0.5	0	0	0	0	0
		[Min, Max]	[0, 4]	[0, 2]	[0, 1]	[0, 0]	[0, 4]	[0, 1]	[0, 3]	[0, 1]	[0, 0]	[0, 3]

Table 3: Median number of refinement actions made by participants in each condition for each dataset (Median [Min, Max]).

		Label			Description	Group			Total
		Add	Delete	Revise	Revise	Add	Delete	Revise	
Natural Scene	Baseline	1 [0, 5]	0 [0, 6]	2.5 [0, 8]	3.5 [0, 9]	0 [0, 1]	0 [0, 1]	0 [0, 1]	8.5 [2, 24]
	DynamicLabels	2 [0, 8]	2 [0, 9]	1.5 [0, 6]	0.5 [0, 2]	0 [0, 1]	0 [0, 1]	0 [0, 0]	7 [0, 25]
Event Flier	Baseline	0.5 [0, 8]	1.5 [0, 5]	3 [1, 9]	1 [0, 7]	0 [0, 3]	0 [0, 4]	0 [0, 0]	9.5 [2, 23]
	DynamicLabels	2.5 [0, 5]	1.5 [0, 8]	1 [0, 5]	3 [0, 11]	0 [0, 1]	0 [0, 2]	0 [0, 0]	11 [5, 15]

Table 4: Average number of crowd feedback collected per single label set (Mean [Min, Max]): number of crowd labels, number of unique crowd labels, number of annotations with crowd-made labels, number of annotations with ML practitioner-made labels, for each dataset and condition.

	Natural Scene		Event flier	
	Baseline	DynamicLabels	Baseline	DynamicLabels
# of crowd labels	-	179.75 [143, 206]	-	199.25 [143, 229]
# of unique crowd labels	-	72.13 [47, 92]	-	106.00 [83, 137]
# of annotations with crowd-made labels	-	969.38 [770, 1026]	-	991.00 [813, 1031]
# of annotations with ML practitioner-made labels	1053.63 [990, 1294]	969.25 [784, 1024]	1023.75 [1021, 1030]	989.63 [812, 1028]

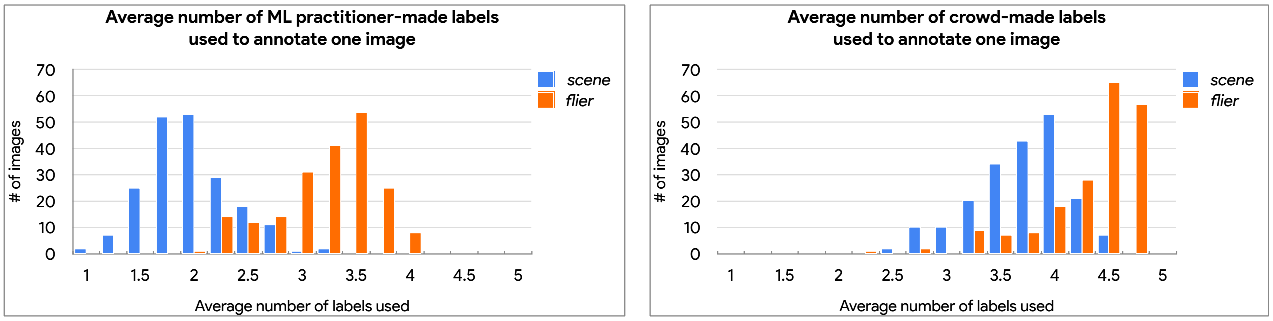


Figure 8: Overall distribution of the average number of labels used to annotate one image. The left chart shows the average number of ML practitioner-made labels used, and the right chart shows the average number of crowd-made labels used for a single image.

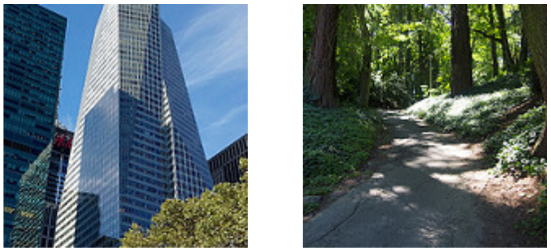


Figure 9: Two example images in the *scene* dataset. Five different labels (civilization , city , town , buildings , metrocity) were made for the left *scene* image consisting of two tall buildings and a sky in the background, and three labels (forest , hills , trees) for the right *scene* image illustrating a trail inside a forest with trees.

6.3 RQ2: How do ML practitioners use crowd feedback to refine their label sets?

Both crowd annotations and crowd-made labels were utilized to help the participants understand and apply the feedback as refinements. The crowd annotations helped ML practitioners to (1) understand the crowd's general opinions and perspectives and (2) spot the weakness of their label sets, and the crowd-made labels helped them (3) explore and apply relevant ones to their label sets.

6.3.1 Understanding the crowd's general opinions and perspectives. Participants mentioned that they could observe both converging and diverging opinions in the crowd feedback. Collective opinions were visible through *label suggestions* or *most common crowd-made labels*, and the participants compared their label set with the crowds' labels to see the similarity of their labels to the general crowd's labels. P3 mentioned "I am usually afraid of bias, especially when building a label set on my own. Seeing that many of the crowd have made similar labels with mine, I am more confident that [this is the] right direction." Further, P16 said that "looking at the most common labels helps to deal with the ambiguity in constructing a label set by oneself, and if there is a label that many crowd workers made, then that shows the necessity of that particular label."

In contrast, participants also observed the diverging opinions of the crowd in the collected annotations and labels. While there were overlapping crowd-made labels, participants found the labels to be overall diverging, which informed them how an image can be perceived differently by the workers. P13 said, "by looking at the crowd annotation results for labels and images I was unsure of in session 1, I am more confident that my label set should be defined more clearly."

6.3.2 Understanding the weakness of their label set. They were also able to realize the weaknesses of their label set through the feedback. This was mostly done by looking at the actual annotated images using their label set. P2 commented, "I would not have realized how poorly built my label set is without looking at the annotation results", and made major changes (adding 3, revising 3, deleting 1 label) to their label set in session 2. A common weakness identified by participants was the lack of good label descriptions, found by looking at the detailed view of each label, and ambiguous boundaries between labels, found by looking at the top conflicts. P9 commented, in *flier*, that they "would not have known that yoga fliers could [be perceived as] the 'nature' category without crowd annotations." They changed the description for nature , from 'Any poster which the main topic is about nature' to 'Any poster about plants, forest, and nature' to better specify and communicate the boundary of the label.

6.3.3 Exploring alternative labels and applying perspectives of the crowd. Participants were also able to incorporate more perspectives from the crowd into their label sets. As DynamicLabels made crowd-made labels more visible throughout the entire refinement process, the participants were able to easily refer to the refinement suggestions, most common labels, or each worker's labels. A big portion of adding/revising refinements (*scene*: 35.1%, *flier*: 29.0%) in DynamicLabels utilized crowd-made labels. On the other hand, in baseline, no participants directly referenced crowd expressions (from 'reasons for others') in adding/revising labels.

Participants were also able to make more satisfactory refinements by referring to the crowd-made labels in DynamicLabels. In baseline, participants eventually got a sense of the labels that needed to be revised by extracting summative information from the annotation results, but the next challenge they faced was in making satisfactory refinements. Participants struggled to come up with satisfactory label names, which led to more label name changes happening in the baseline system. For example, in the baseline, P10 revised a single label three times, from snow to snow/glacier without mountain , extreme cold with snow and glacier , then to extreme cold with snow, glaciers, and mountains and explicitly said that the crowd-made labels would have been helpful to decide the label name.

As a result, participants made more changes in label name in the baseline (with a median of 2.5 for *scene* 3 for *flier*) than in the DynamicLabels (with a median of 1.5 for *scene* and 1 for *flier*) The difference between conditions is statistically significant (two-way ART ANOVA [12, 31], F=5.34 with $p<0.05$).

6.4 RQ3: How do ML practitioners use the refinement interface in DynamicLabels to make informed refinement decisions?

Throughout the study, we were able to observe the distinctive benefits of DynamicLabels over the baseline system in making more informed refinement decisions. When asked to compare the two conditions on how they helped their refinement decisions, most participants (13/16) rated DynamicLabels better than baseline.

The results state that DynamicLabels supports (1) a high-level understanding of the feedback with metrics, (2) better assurance through examining multiple options, (3) a structured refinement process. In addition, DynamicLabels (4) encouraged a more flexible refinement and (5) surfaced issues that might have been missed in comparison to the baseline. Such benefits supported participants to make more confident, efficient refinements with crowd feedback.

6.4.1 Metrics support understanding of and refinement from the feedback. The most frequently identified strength of DynamicLabels was the existence of the metrics (coverage, conflict) in the Overview component. Participants liked how the metrics summarized the collected annotations and described the metrics as an efficient and intuitive way to understand their label set without looking at the raw data. P1 mentioned that, in the baseline, they had to make much more judgments by themselves, such as understanding the reason behind images with no majority winner, deciding whether to change the label or not by estimating the expected effect of the change, and verifying whether the changes can fix the issues by going through the images again.

Among the metrics, the participants particularly found the conflict metric helpful and many (10/16) aimed at reducing the number of conflicts during the refinement task. They complimented the intuitiveness of the metric, in that "it intuitively shows the labels that are controversial to the crowd (P5)" and utilized the metric to identify which refinements should be prioritized. When a particular label existed in all three top conflicts (e.g., career & socializing, career & volunteering, sports & career for P9 in *flier*), the participants realized that the label can be confusing to the crowd and began their refinement by clarifying and examining the feedback from that label (e.g., career).

6.4.2 Examining and experimenting with various refinement options. With DynamicLabels, participants were able to examine various refinement options before making a decision. With the consequence modal in DynamicLabels that shows the expected changes in the metrics for each refinement action, participants were able to examine each refinement action they considered before applying it. Participants found this consequence modal helpful, as knowing the expected change in the overview helped them make the decision more confidently. In addition, P13 mentioned that "the (action consequence) modal prevented them from making a wrong refinement choice." When P13 tried to replace the label manmade for the conflict between the labels city and countryside, they saw the rise in the conflict and decided to take back the decision. They later merged the two labels into the label manmade. Even P3, who made no refinements with DynamicLabels, examined two refinements they considered but decided not to apply them.

6.4.3 Establishing a structured refinement process. The refinement process with DynamicLabels was perceived to be more structured than with the baseline. With DynamicLabels, participants began their refinement process from the overview, then looked into the detailed view for further understanding, and referred to the crowd-made labels whenever they needed more assurance or references when making refinement decisions. Meanwhile, with the baseline, participants went back and forth between the label view and the image view until they identified the need for change. P6 noted that "in [v1 (baseline)], deciding on the starting or ending points was very challenging as I have to check the image and the annotations repeatedly to understand the outcome of the annotations.". The participants found this implicitly conveyed workflow helpful, as they were able to "prioritize the refinement decisions (P15)." P4 also described DynamicLabels as supporting a more structured process that he could follow to figure out if it was the boundary of the label or the label name that needed to be changed, resulted in a quicker refinement with similar confidence.

6.4.4 Encouraging flexible refinement. Participants noted that having various forms of crowd feedback in DynamicLabels helped them understand the relationship between labels, such as potential conflict or inclusion among labels. During the refinement session, P5 said that "by seeing the number of conflicts between expo and social and going through images with the conflict, I decided to split those labels into more [specific] ones".

Participants also noted that with DynamicLabels, they could focus on how their label set represents the data, whereas they focused on clarifying each label and description in the baseline system. For example, P16 made three merge refinements (e.g., merging city street and buildings into city) with DynamicLabels whereas no high-level refinements were made with the baseline. P13 also mentioned that "[they] would not have combined city and countryside if [they] had not seen the label suggestion manmade after selecting both labels.

With both datasets, more participants made at least one high-level refinement in the DynamicLabels condition (6 out of 8) than in the baseline condition (3 out of 8). With both dataset, the median number of high-level refinements made by participants was 0 (min: 0, max: 3) in the baseline and 1 (min: 0, max: 5) in the DynamicLabels Table 5 summarizes the number of participants who made the split and merge refinement(s) in each condition and dataset.

Table 5: Number of participants who made high-level refinements in each condition for each dataset

	Natural Scene			Event Flier		
	Split	Merge	Total	Split	Merge	Total
Baseline	2	1	3	1	3	3
DynamicLabels	2	5	6	3	4	6

6.4.5 Surfacing conflicts and edge cases that might have been overlooked. In addition to the refinements made, the participants were also able to spot possible conflicts and edge cases that they might have overlooked. When refining with the baseline, most participants made refinements centered around the issues that they expected. P4 mentioned, "I checked that the labels that I assumed to be problematic actually had issues by looking at the annotations, and only revised those labels." However, when refining with DynamicLabels, participants identified unexpected conflicts, and were able to understand where the conflicts were coming from and make suitable changes. Sometimes, participants appreciated the surfacing of unexpected issues even when they did not make refinements accordingly. P7 commented "If there were no crowd data, I would have made refinements solely based on my subjective opinions. Even if I did not reflect all crowd opinions, being able to see them helps me understand potential issues better."

In DynamicLabels, P4 also created more labels by looking at individual crowd-made labels, commenting that "the crowd helped in detecting edge cases in the 200 images." They added the labels cave and desert at the end of their refinement, after seeing images annotated with these labels and realizing their current label set couldn't cover them.

7 DISCUSSION

7.1 Potential use of DynamicLabels in different domains

We believe that DynamicLabels can be generally expanded to domains that do not require special expertise, given the assumption that most crowd workers are the general public. Among them, we suggest a few domains where the benefit could be further amplified. For subjective domains where rules are decided based on collective human judgments (e.g., sentiment classification (P3)), the ML practitioner can effectively understand the general crowd's converging opinions and identify a convincing distinction between labels. For complicated domains where having a large number of labels is necessary (e.g., receipt information extraction (P15)), the practitioner can use DynamicLabels to identify potential edge cases. If they have access to a group of domain experts, they could utilize DynamicLabels for a comprehensive understanding of the data with experts' opinions. For example, P7 mentioned that if they could use DynamicLabels with a group of graduate students, they want to try label set building for topic classification of research papers.

7.2 Providing various forms of crowd feedback and giving more control to ML practitioners over them

In DynamicLabels, we present crowd feedback in various aspects through *overview*, *detailed view*, and *crowd label view*). In our study, ML practitioners flexibly utilized these features in combination to meet their needs, which can change over the process of understanding issues and the evidence behind them. At the same time, some participants expressed the need to see raw crowd feedback, such as raw annotations of each image and crowd-made labels and annotations before post-processing. Seeing that the participants made use of multiple features of their choice and were not reluctant to examine more data with increasing complexity, presenting them with a more dynamic version of analyses that covers a wide variety of crowd feedback will be useful in making confident refinements.

Also, while the collected feedback was provided to the ML practitioners to explore, participants expressed the need for more direct control over the collected feedback to reflect in the analysis. For example, participants wanted to filter out trolling workers' feedback or give higher weights to certain workers to explore how the conflict changes with more reliable workers' opinions valued. Inspired by Jury Learning's approach [14] in allowing the model builder to compose a group of juries and their opinions, DynamicLabels can have ML practitioners focus on a particular group's perspectives to construct the label set. Participants also wished to have more control over the crowd annotations by fixing annotations they thought were incorrect to indicate that particular labels or images had been examined. With this control, ML practitioners can make sure that all feedback is considered and applied.

7.3 Expanding the support in DynamicLabels for more advanced, automated label set refinements

In the current design of DynamicLabels, the crowd generates their feedback solely on their own, and refinement supports primarily rely on the ML practitioners' ability to make the final decision, such as in the action consequence modal where the ML practitioner needs to confirm the action. As our goal was to examine the end-to-end effect of label set refinement with crowd feedback, we intentionally chose a basic form of support for less complexity, which could be further strengthened through existing approaches (e.g., LLMs and visual analytics) and long-term adaptation.

To support a more streamlined *crowd feedback* generation process, we can utilize large-language models for their ability to generate and synthesize to streamline the process. LLMs can collaborate with the crowd to help alleviate the burden of having to generate labels and annotate all assigned data by recommending labels or providing initial annotation for the crowd to correct. This could support the crowd to focus on generating helpful feedback. However, this must be carefully designed so that the initial feedback of the crowd is not hindered by LLM's work [22].

In supporting an informed *refinement decision*, the key is to optimize between making an *accurate* decision and an *efficient* decision. For supporting accurate decisions, annotations that need to be re-annotated with the change of label could be automatically detected and reflected, by using an approach similar to the error detection approach in Bäuerle et al. [5] or LabelVizier [38]. For supporting efficient decisions, we imagine an additional module where the user's exploration focus from provided visual analytics could be more easily reflected in refinement suggestions. For example, if the ML practitioner aims to minimize conflicting labels for their task, the visual exploration could focus on detecting overlaps or potential conflicts. For long-term or larger-scale use cases, the suggestions could be made adaptive in terms of the core metrics and important aspects of the data, as previous work [20, 25] emphasizes the benefit of having diverse types of analyses for contextualized support for different data and task type.

7.4 Designing a more human-centered model with the crowd

As the label set constitutes the primary structure of the model, aligning it with the user's mental model earlier in the label set construction stage can be effective in incorporating the potential user's mental model into the model. An interesting transition of the participants' behaviors we saw in session 2 was that they were considering and reflecting more on the crowd's opinions, contrary to session 1 where they focused on creating a clear distinction among the labels for better model training. P12 commented, "In session 1, I mainly built my label set considering the ML model by focusing on the semantic aspects of the images. However, during session 2, I additionally considered the annotators, and ended up splitting the labels more and grouping the labels for easier understanding. P12 was worried about the task becoming difficult for the model, but still thought of this consideration of the user valuable.

We believe that DynamicLabels's impact can be enhanced far beyond building label sets with the crowd opinion, but further in incorporating the crowd’s—or the real users’—opinions in the process of building machine learning applications. For example, in the dataset construction stage, the crowd can provide additional explanations or rationales for each annotation, which can be utilized to train the model to generate more human-like explanations or logic in a similar way to humans. In addition, crowd feedback can be used to collect large-scale opinions about the performance of large models and to come up with human-centered metrics to evaluate those models. When crowd feedback is utilized in the later stages of model building (e.g., model building, model evaluation), the crowd can naturally learn about how the ML model functions while providing feedback, which enables a natural human-AI interaction.

8 LIMITATIONS & FUTURE WORK

We acknowledge several limitations of this work and discuss possible future work.

System design. While we applied quality control methods by providing the crowd workers with tutorial tasks or warnings on poor work, the accuracy of the collected feedback was lower than the ML practitioners’ expectations. Subsequently, some participants lost trust in the crowd feedback, hindering them from actively utilizing the provided interface or ending up making passive refinements. We believe that additional quality control methods such as an attention-checking task with a gold standard [16] could improve the feedback quality. Further, a more complex crowdsourcing workflow such as worker deliberation methods (as proven effective in [27]) to identify which specific cases are more diverse or require additional attention could help the ML practitioner prioritize the core issues.

We also did not incorporate any machine learning approaches in post-processing or generating the analyses of the review interface, as they caused latency in the system. Future works could incorporate such approaches and investigate how to optimally utilize automatic techniques for a more efficient yet well-thought-out label set construction.

Study setting. In the study, we did not communicate the purpose of the dataset or the model to the crowd, which would have led to the feedback being inherently diverse. Including that information would have resulted in higher quality and relevant feedback, but also may have hindered the feedback from containing fresh and natural perspectives. Investigating task designs to embed a clear purpose of the ML practitioner but also extract fresh and diverse feedback could be valuable as the next steps of this research. To measure the effect of crowd feedback and the label set refinement interface, we conducted a comparison study with many factors (e.g., the number of images used, amount of crowd feedback) controlled throughout. Since participants exhibited different patterns in utilizing crowd feedback, we wish to observe the long-term effects of DynamicLabels by testing the system in a more realistic setting.

In addition, the study focused on investigating whether DynamicLabels successfully supported an informed label set construction process, and did not validate the quality of the label set. While the ML practitioners were satisfied with the changes they made, a larger, longer-term study with more iterations to investigate the outcome of the workflow and its performance in the later processes of the ML model construction is a future step.

9 CONCLUSION

In this paper, we present DynamicLabels, a system that supports the process of label set construction with crowd feedback and interactive feedback analysis. Our study with 16 participants shows that DynamicLabels enables a more exploratory, flexible, and structured refinement process with fine-grained analysis and crowd-made labels. The crowd feedback helped the participants understand the general crowd’s opinions and the weaknesses of their label set, and actively utilized the crowd-made labels for refinements. DynamicLabels suggests a new approach to building label sets for ML models, by incorporating the crowd’s—or the general user’s—feedback to reflect the user’s diverse opinions and perspectives.

ACKNOWLEDGMENTS

This work was supported by NAVER CLOVA and the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2021-0-01347, Video Interaction Technologies Using Object-Oriented Video Modeling). The authors would like to thank the study participants for their insightful comments, the members of KIXLAB and the reviewers for their thorough feedback over the years to improve this work.

REFERENCES

[1] David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *the Journal of machine Learning research* 3 (2003), 993–1022.

[2] 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, 1 (Nov. 2013), 25–33. https://doi.org/10.1609/hcomp.v1i1.13091

[3] Marco Brambilla, Jordi Cabot, Javier Luis Cánovas Izquierdo, and Andrea Mauri. 2017. Better call the crowd: using crowdsourcing to shape the notation of domain-specific languages. In *Proceedings of the 10th ACM SIGPLAN International Conference on Software Language Engineering* (Vancouver, BC, Canada) *(SLE 2017)*. Association for Computing Machinery, New York, NY, USA, 129–138. https://doi.org/10.1145/3136014.3136033

[4] Magdalena Bujar, Neil McAuslane, Stuart Walker, and Sam Salek. 2022. A process for evaluating quality decision-making practices during the development, review and reimbursement of medicines. *International Journal of Health Policy and Management* 11, 2 (2022), 128.

[5] A. Bäuerle, H. Neumann, and T. Ropinski. 2020. Classifier-Guided Visual Correction of Noisy Labels for Image Classification Tasks. *Computer Graphics Forum* 39, 3 (2020), 195–205. https://doi.org/10.1111/cgf.13973 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.13973

[6] Carrie J. Cai, Emily Reif, Narayan Hegde, Jason Hipp, Been Kim, Daniel Smilkov, Martin Wattenberg, Fernanda Viegas, Greg S. Corrado, Martin C. Stumpe, and Michael Terry. 2019. Human-Centered Tools for Coping with Imperfect Algorithms During Medical Decision-Making. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI ’19)*. Association for Computing Machinery, New York, NY, USA, 1–14. https://doi.org/10.1145/3290605.3300234

[7] Joseph Chee Chang, Saleema Amershi, and Ece Kamar. 2017. Revolt: Collaborative Crowdsourcing for Labeling Machine Learning Datasets. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI ’17)*. Association for Computing Machinery, New York, NY, USA, 2334–2346. https://doi.org/10.1145/3025453.3026044

[8] Joseph Chee Chang, Aniket Kittur, and Nathan Hahn. 2016. Alloy: Clustering with Crowds and Computation. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI ’16)*. Association for Computing Machinery, New York, NY, USA, 3180–3191. https://doi.org/10.1145/2858036.2858411

[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*. Association for Computing Machinery, New York, NY, USA, 1999–2008. https://doi.org/10.1145/2470654.2466265

[10] Yoonseo Choi, Toni-Jan Keith Palma Monserrat, Jeongeon Park, Hyungyu Shin, Nyoungwoo Lee, and Juho Kim. 2021. Protochat: Supporting the conversation

design process with crowd feedback. *Proceedings of the ACM on Human-Computer Interaction* 4, CSCW3 (2021), 1–27.

[11] Google Cloud. 2019. Creating label sets. https://cloud.google.com/ai-platform/data-labeling/docs/label-sets. Accessed: 2023-01-15.

[12] Lisa A. Elkin, Matthew Kay, James J. Higgins, and Jacob O. Wobbrock. 2021. An Aligned Rank Transform Procedure for Multifactor Contrast Tests. In *The 34th Annual ACM Symposium on User Interface Software and Technology* (Virtual Event, USA) *(UIST '21)*. Association for Computing Machinery, New York, NY, USA, 754–768. https://doi.org/10.1145/3472749.3474784

[13] Elena L. Glassman, Juho Kim, Andrés Monroy-Hernández, and Meredith Ringel Morris. 2015. Mudslide: A Spatially Anchored Census of Student Confusion for Online Lecture Videos. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (Seoul, Republic of Korea) *(CHI '15)*. Association for Computing Machinery, New York, NY, USA, 1555–1564. https://doi.org/10.1145/2702123.2702304

[14] Mitchell L. Gordon, Michelle S. Lam, Joon Sung Park, Kayur Patel, Jeff Hancock, Tatsunori Hashimoto, and Michael S. Bernstein. 2022. Jury Learning: Integrating Dissenting Voices into Machine Learning Models. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (<conf-loc>, <city>New Orleans</city>, <state>LA</state>, <country>USA</country>, </conf-loc>) *(CHI '22)*. Association for Computing Machinery, New York, NY, USA, Article 115, 19 pages. https://doi.org/10.1145/3491102.3502004

[15] Kosa Goucher-Lambert and Jonathan Cagan. 2019. Crowdsourcing inspiration: Using crowd generated inspirational stimuli to support designer ideation. *Design Studies* 61 (2019), 1–29.

[16] Danula Hettiachchi, Mike Schaekermann, Tristan J McKinney, and Matthew Lease. 2021. The challenge of variable effort crowdsourcing and how visible gold can help. *Proceedings of the ACM on Human-Computer Interaction* 5, CSCW2 (2021), 1–26.

[17] Md Naimul Hoque, Wenbin He, Arvind Kumar Shekar, Liang Gou, and Liu Ren. 2023. Visual Concept Programming: A Visual Analytics Approach to Injecting Human Intelligence at Scale. *IEEE Transactions on Visualization and Computer Graphics* 29, 1 (2023), 74–83. https://doi.org/10.1109/TVCG.2022.3209466

[18] Sanjay Kairam and Jeffrey Heer. 2016. Parting Crowds: Characterizing Divergent Interpretations in Crowdsourced Annotation Tasks. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing* (San Francisco, California, USA) *(CSCW '16)*. Association for Computing Machinery, New York, NY, USA, 1637–1648. https://doi.org/10.1145/2818048.2820016

[19] Juho Kim, Haoqi Zhang, Paul André, Lydia B. Chilton, Wendy Mackay, Michel Beaudouin-Lafon, Robert C. Miller, and Steven P. Dow. 2013. Cobi: a community-informed conference scheduling tool. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology* (St. Andrews, Scotland, United Kingdom) *(UIST '13)*. Association for Computing Machinery, New York, NY, USA, 173–182. https://doi.org/10.1145/2501988.2502034

[20] Shixia Liu, Changjian Chen, Yafeng Lu, Fangxin Ouyang, and Bin Wang. 2019. An Interactive Method to Improve Crowdsourced Annotations. *IEEE Transactions on Visualization and Computer Graphics* 25, 1 (2019), 235–245. https://doi.org/10.1109/TVCG.2018.2864843

[21] Kurt Luther, Jari-Lee Tolentino, Wei Wu, Amy Pavel, Brian P. Bailey, Maneesh Agrawala, Björn Hartmann, and Steven P. Dow. 2015. Structuring, Aggregating, and Evaluating Crowdsourced Design Critique. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing* (Vancouver, BC, Canada) *(CSCW '15)*. Association for Computing Machinery, New York, NY, USA, 473–485. https://doi.org/10.1145/2675133.2675283

[22] Qianou Ma, Tongshuang Wu, and Kenneth Koedinger. 2023. Is AI the better programming partner? Human-Human Pair Programming vs. Human-AI pAIr Programming. arXiv:2306.05153 [cs.HC]

[23] Michael Muller, Christine T. Wolf, Josh Andres, Michael Desmond, Narendra Nath Joshi, Zahra Ashktorab, Aabhas Sharma, Kristina Brimijoin, Qian Pan, Evelyn Duesterwald, and Casey Dugan. 2021. Designing Ground Truth and the Social Life of Labels. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 94, 16 pages. https://doi.org/10.1145/3411764.3445402

[24] Deokgun Park, Seungyeon Kim, Jurim Lee, Jaegul Choo, Nicholas Diakopoulos, and Niklas Elmqvist. 2017. Conceptvector: text visual analytics via interactive lexicon building using word embedding. *IEEE transactions on visualization and computer graphics* 24, 1 (2017), 361–370.

[25] Ji Hwan Park, Saad Nadeem, Saeed Boorboor, Joseph Marino, and Arie Kaufman. 2021. CMed: Crowd Analytics for Medical Imaging Data. *IEEE Transactions on Visualization and Computer Graphics* 27, 6 (2021), 2869–2880. https://doi.org/10.1109/TVCG.2019.2953026

[26] Nithya Sambasivan, Shivani Kapania, Hannah Highfill, Diana Akrong, Praveen Paritosh, and Lora M Aroyo. 2021. "Everyone wants to do the model work, not the data work": Data Cascades in High-Stakes AI. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 39, 15 pages. https://doi.org/10.1145/3411764.3445518

[27] Mike Schaekermann, Joslin Goh, Kate Larson, and Edith Law. 2018. Resolvable vs. irresolvable disagreement: A study on worker deliberation in crowd work. *Proceedings of the ACM on Human-Computer Interaction* 2, CSCW (2018), 1–19.

[28] Shahin Sharifi Noorian, Sihang Qiu, Burcu Sayin, Agathe Balayn, Ujwal Gadiraju, Jie Yang, and Alessandro Bozzon. 2023. Perspective: Leveraging Human Understanding for Identifying and Characterizing Image Atypicality. In *Proceedings of the 28th International Conference on Intelligent User Interfaces* (Sydney, NSW, Australia) *(IUI '23)*. Association for Computing Machinery, New York, NY, USA, 650–663. https://doi.org/10.1145/3581641.3584096

[29] Selina Sutton and Shaun Lawson. 2017. A Provocation for Rethinking and Democratising Emoji Design. In *Proceedings of the 2017 ACM Conference Companion Publication on Designing Interactive Systems* (Edinburgh, United Kingdom) *(DIS '17 Companion)*. Association for Computing Machinery, New York, NY, USA, 7–12. https://doi.org/10.1145/3064857.3079109

[30] Wesley Willett, Shiry Ginosar, Avital Steinitz, Björn Hartmann, and Maneesh Agrawala. 2013. Identifying Redundancy and Exposing Provenance in Crowdsourced Data Analysis. *IEEE Transactions on Visualization and Computer Graphics* 19, 12 (2013), 2198–2206. https://doi.org/10.1109/TVCG.2013.164

[31] Jacob O. Wobbrock, Leah Findlater, Darren Gergle, and James J. Higgins. 2011. The aligned rank transform for nonparametric factorial analyses using only anova procedures. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (<conf-loc>, <city>Vancouver</city>, <state>BC</state>, <country>Canada</country>, </conf-loc>) *(CHI '11)*. Association for Computing Machinery, New York, NY, USA, 143–146. https://doi.org/10.1145/1978942.1978963

[32] Yingcai Wu, Furu Wei, Shixia Liu, Norman Au, Weiwei Cui, Hong Zhou, and Huamin Qu. 2010. OpinionSeer: interactive visualization of hotel customer feedback. *IEEE transactions on visualization and computer graphics* 16, 6 (2010), 1109–1118.

[33] Anbang Xu, Shih-Wen Huang, and Brian Bailey. 2014. Voyant: generating structured feedback on visual designs using a crowd of non-experts. In *Proceedings of the 17th ACM Conference on Computer Supported Cooperative Work & Social Computing* (Baltimore, Maryland, USA) *(CSCW '14)*. Association for Computing Machinery, New York, NY, USA, 1433–1444. https://doi.org/10.1145/2531602.2531604

[34] Guorong Xuan, Wei Zhang, and Peiqi Chai. 2001. EM algorithms of Gaussian mixture model and hidden Markov model. In *Proceedings 2001 International Conference on Image Processing (Cat. No.01CH37205)*, Vol. 1. IEEE, Thessaloniki, Greece, 145–148 vol.1. https://doi.org/10.1109/ICIP.2001.958974

[35] W. Yang, X. Ye, X. Zhang, L. Xiao, J. Xia, Z. Wang, J. Zhu, H. Pfister, and S. Liu. 2022. Diagnosing Ensemble Few-Shot Classifiers. *IEEE Transactions on Visualization and Computer Graphics* 28, 09 (sep 2022), 3292–3306. https://doi.org/10.1109/TVCG.2022.3182488

[36] Yu-Chun Grace Yen, Joy O. Kim, and Brian P. Bailey. 2020. Decipher: An Interactive Visualization Tool for Interpreting Unstructured Design Feedback from Multiple Providers. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (<conf-loc>, <city>Honolulu</city>, <state>HI</state>, <country>USA</country>, </conf-loc>) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3313831.3376380

[37] Amy X. Zhang, Jilin Chen, Wei Chai, Jinjun Xu, Lichan Hong, and Ed CHI. 2018. Evaluation and Refinement of Clustered Search Results with the Crowd. *ACM Trans. Interact. Intell. Syst.* 8, 2, Article 14 (jun 2018), 28 pages. https://doi.org/10.1145/3158226

[38] X. Zhang, X. Xuan, A. Dima, T. Sexton, and K. Ma. 2023. LabelVizier: Interactive Validation and Relabeling for Technical Text Annotations. In *2023 IEEE 16th Pacific Visualization Symposium (PacificVis)*. IEEE Computer Society, Los Alamitos, CA, USA, 167–176. https://doi.org/10.1109/PacificVis56936.2023.00026

[39] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. 2018. Places: A 10 Million Image Database for Scene Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40, 6 (2018), 1452–1464. https://doi.org/10.1109/TPAMI.2017.2723009

A APPENDIX

A.1 Participant Demographics

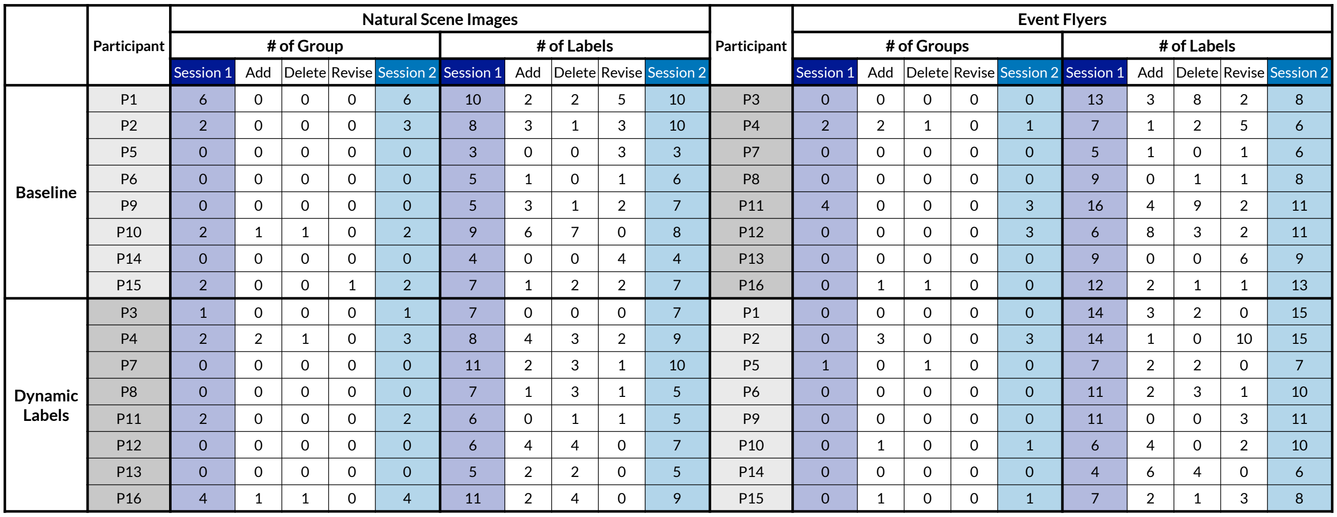
In Table 6, we describe detailed demographics for each participant.

Table 6: Detailed demographic information of each participant including occupation, prior domain of experience, and task order with dataset, condition, and task defined. We provided the two datasets to the users and asked them to identify a task before the beginning of Session 1. For occupation, Grad indicates graduate student, Undergrad indicates undergraduate student, and Industry indicates industry workers.

PID (Occupation)	Prior domain of experience	Task Order: Dataset (Condition, Task Defined)
P1 (Grad)	Action Classification	(1) Scene (*Baseline*, Background video generation from input script) → (2) Flier (*DynamicLabels*, Events suggestions from user preference)
P2 (Industry)	Image Segmentation	(1) Scene (*Baseline*, Display scene image which suits the atmosphere) → (2) Flier (*DynamicLabels*, Reference searching for flier creators)
P3 (Industry)	Sentence Structure Classification	(1) Flier (*Baseline*, Searching for similar events through category) → (2) Scene (*DynamicLabels*, Searching for similar scene images)
P4 (Undergrad)	Footprint Classification	(1) Scene (*DynamicLabels*, Searching for similar scene images) → (2) Flier (*Baseline*, Reference searching for design elements)
P5 (Industry)	Text Classification; Stress Level Prediction	(1) Flier (*DynamicLabels*, Events suggestions from user preference) → (2) Scene (*Baseline*, Recommending computer wallpapers)
P6 (Grad)	Sensor Data Classification; Speech Data Classification	(1) Scene (*Baseline*, Scene detection for automatic image editing) → (2) Flier (*DynamicLabels*, Searching for similar events through category)
P7 (Grad)	Topic Evaluation Classification	(1) Flier (*Baseline*, Events suggestions from user preference) → (2) Scene (*DynamicLabels*, Automatic categorization of captured images)
P8 (Grad)	Video Classification; Bounding Box Detection	(1) Scene (*DynamicLabels*, Data for text-to-image generation model) → (2) Flier (*Baseline*, Estimating cost for flier characteristics)
P9 (Undergrad)	Image Classification	(1) Flier (*DynamicLabels*, Recommending other events through category) → (2) Scene (*Baseline*, Searching for similar scene images)
P10 (Undergrad)	Tweet Emotion Classification	(1) Flier (*DynamicLabels*, Searching for similar events through category) → (2) Scene (*Baseline*, Location detection from scene images)
P11 (Grad)	Pattern (Trajectory) Classification	(1) Flier (*Baseline*, Searching for similar events through category) → (2) Scene (*DynamicLabels*, Location detection from scene images)
P12 (Industry)	Diagnosis Classification	(1) Flier (*Baseline*, Searching for similar events through category) → (2) Scene (*DynamicLabels*, Region identification from scene images)
P13 (Industry)	Emotion Classification; Video Classification	(1) Scene (*DynamicLabels*, Region identification from scene images) → (2) Flier (*Baseline*, Interpretation of events with text and image)
P14 (Grad)	Disease Classification Task; Mask Detection	(1) Scene (*Baseline*, Recommending computer wallpapers) → (2) Flier (*DynamicLabels*, Searching for similar events through category)
P15 (Grad)	Emotion Classification	(1) Flier (*DynamicLabels*, Searching for similar events through category) → (2) Scene (*Baseline*, Categorization for scene image searching)
P16 (Industry)	Text Intention Classification	(1) Scene (*DynamicLabels*, Region identification from scene images) → (2) Flier (*Baseline*, Categorization of fliers for advertisement display)

A.2 Label Set Refinement Logs

In the below table (Figure 10, we describe the number of groups and labels for each participant in sessions 1 and 2, and the refinement actions they made in between.



	Participant	Natural Scene Images										Participant	Event Flyers									
		# of Group					# of Labels						# of Groups					# of Labels				
		Session 1	Add	Delete	Revise	Session 2	Session 1	Add	Delete	Revise	Session 2		Session 1	Add	Delete	Revise	Session 2	Session 1	Add	Delete	Revise	Session 2
Baseline	P1	6	0	0	0	6	10	2	2	5	10	P3	0	0	0	0	0	13	3	8	2	8
	P2	2	0	0	0	3	8	3	1	3	10	P4	2	2	1	0	1	7	1	2	5	6
	P5	0	0	0	0	0	3	0	0	3	3	P7	0	0	0	0	0	5	1	0	1	6
	P6	0	0	0	0	0	5	1	0	1	6	P8	0	0	0	0	0	9	0	1	1	8
	P9	0	0	0	0	0	5	3	1	2	7	P11	4	0	0	0	3	16	4	9	2	11
	P10	2	1	1	0	2	9	6	7	0	8	P12	0	0	0	0	3	6	8	3	2	11
	P14	0	0	0	0	0	4	0	0	4	4	P13	0	0	0	0	0	9	0	0	6	9
	P15	2	0	0	1	2	7	1	2	2	7	P16	0	1	1	0	0	12	2	1	1	13
Dynamic Labels	P3	1	0	0	0	1	7	0	0	0	7	P1	0	0	0	0	0	14	3	2	0	15
	P4	2	2	1	0	3	8	4	3	2	9	P2	0	3	0	0	3	14	1	0	10	15
	P7	0	0	0	0	0	11	2	3	1	10	P5	1	0	1	0	0	7	2	2	0	7
	P8	0	0	0	0	0	7	1	3	1	5	P6	0	0	0	0	0	11	2	3	1	10
	P11	2	0	0	0	2	6	0	1	1	5	P9	0	0	0	0	0	11	0	0	3	11
	P12	0	0	0	0	0	6	4	4	0	7	P10	0	1	0	0	1	6	4	0	2	10
	P13	0	0	0	0	0	5	2	2	0	5	P14	0	0	0	0	0	4	6	4	0	6
	P16	4	1	1	0	4	11	2	4	0	9	P15	0	1	0	0	1	7	2	1	3	8

Figure 10: Number of groups and labels in sessions 1 and session 2 for each participant in each data type, and the number of groups and labels added/deleted/revised in between the sessions.

A.3 Example Label Set Refinement

Figure 11 demonstrates P15's initial and final label set, as well as the refinements.

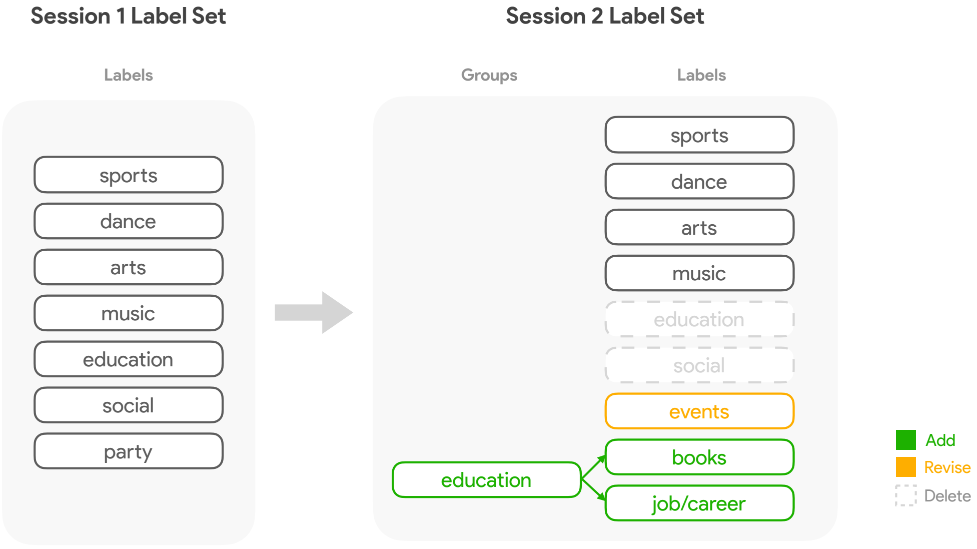


Figure 11: An illustrative example of how one participant's label set changed in sessions 1 and 2 (P15). Two labels (education, social) are deleted, a label is revised (from social to events), and two labels (books, job/career) are added and grouped with a newly added group (education)