Security Protection and Quality Control in Crowdsourcing

CAE Tech Talk - Thursday, January 20th, 2022

Presenters: **Weiping Pei**, Chuan Yue

COLORADO SCHOOL OF MINES
engineering the way

Outline

1. Introduction of Crowdsourcing and Data Quality

2. Attack on Attention Check Mechanism: AC-EasyPass
   - "Attention Please: Your Attention Check Questions in Survey Studies Can Be Automatically Answered", The Web Conference (WWW), 2020

3. Fine-grained Behavior-based Quality Control (FBQC)
   - "Quality Control in Crowdsourcing based on Fine-Grained Behavioral Features", ACM Conference on Computer-Supported Cooperative Work and Social Computing (CSCW), 2021

4. Conclusion

1. Introduction

Importance and Impact of Crowdsourcing

Crowdsourcing systems (e.g., Amazon Mechanical Turk, Appen) leverage the wisdom of crowds to facilitate data collection and annotation/labeling: for **researchers or decision makers** in many disciplines and for **AI model designers or developers**.

(1) Consumer Research

In the Journal of Consumer Research (June 2015–April 2016), 43% of behavioral studies were conducted on the crowdsourcing system Amazon Mechanical Turk (MTurk).

(2) Social Science Research

In social science journals with an impact factor greater than 2.5, 2011 saw fewer than 50 papers using data from MTurk, whereas 2015 saw more than 500.

(3) Large-scale Datasets in Machine Learning

ImageNet (3.2 M images)
Open Image (16 M bounding boxes on 1.9 M images)
MS COCO (2.5 M instances on 328 K images)
SQuAD (100 K questions on 536 article)
SST (215 K on phrases on 11.8 K sentences)

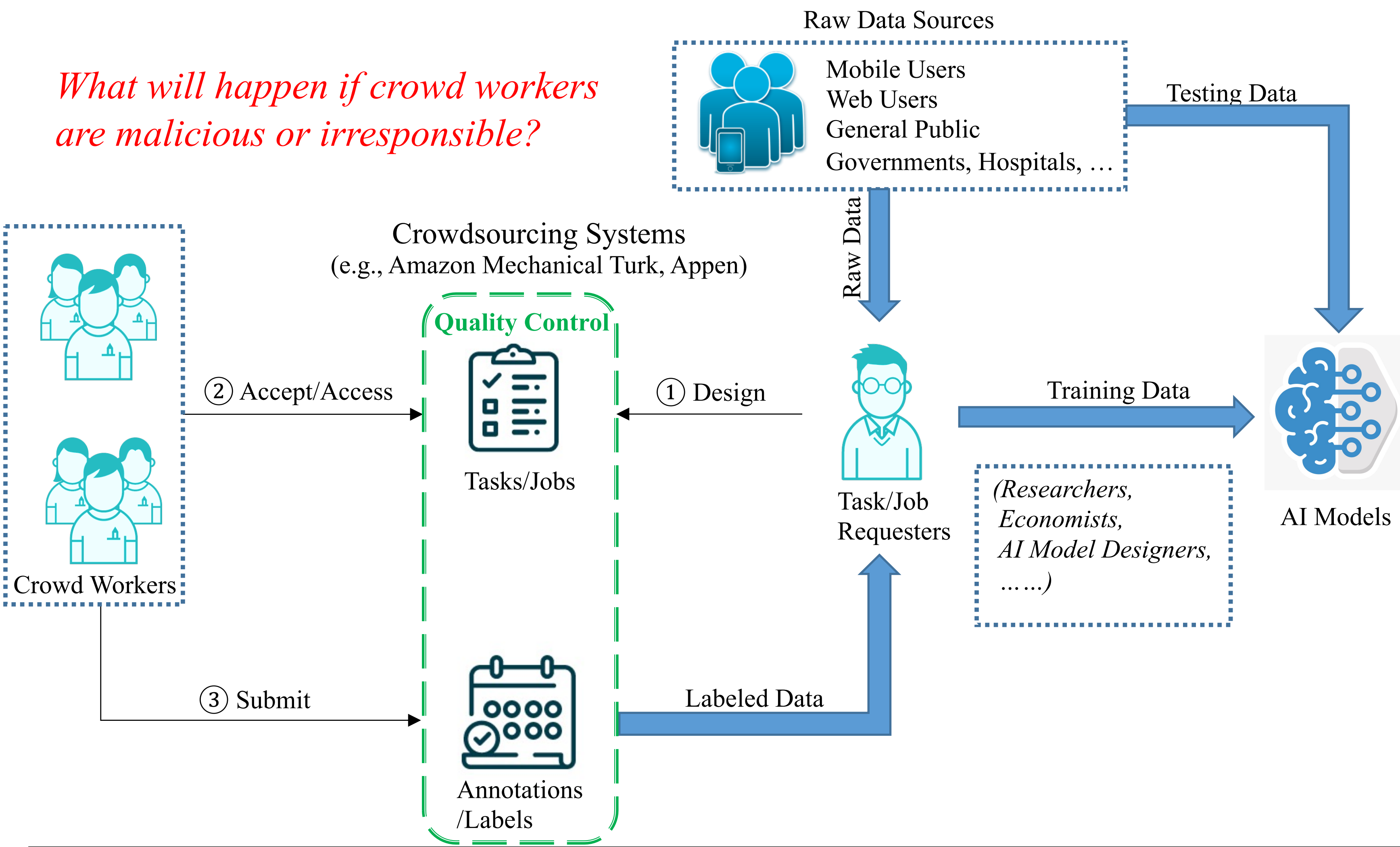
(4) Real-world Applications for People with Visual Impairments

Be My Eye
Aira

1. Introduction

Overview of Crowdsourcing Systems and Stakeholders

What will happen if crowd workers are malicious or irresponsible?



1. Introduction

Consequences of Low-quality or Manipulated Data

(1) A research survey of public opinions (e.g., pre-election or COVID related polls)
Irresponsible workers: randomly select answers
Malicious workers: always select negative responses (e.g., "strongly disagree")
Incorrect or manipulated research results; misleading information to the public

(2) Large-scale Dataset Collection for Machine Learning
Irresponsible workers: carelessly provide annotations or lack of necessary skills
Malicious workers: provide wrong annotations on purpose
Poor performance or pollution (poisoning) in trained AI models

A Bot Panic Hits Amazon's Mechanical Turk [1]

In August 2018, MTurk had a "bot" panic: psychology researchers have noticed a spike in poor quality survey responses collected on MTurk.

[1] https://www.wired.com/story/amazon-mechanical-turk-bot-panic/

1. Introduction

Research Questions

- What are potential vulnerabilities and risks that could compromise data quality and integrity in crowdsourcing systems?

- How can we mitigate risks and prevent attacks to ensure data quality and integrity?

Outline

1. Introduction of Crowdsourcing and Data Quality

2. Attack on Attention Check Mechanism: AC-EasyPass
    - "Attention Please: Your Attention Check Questions in Survey Studies Can Be Automatically Answered", The Web Conference (WWW), 2020

3. Fine-grained Behavior-based Quality Control (FBQC)
    - "Quality Control in Crowdsourcing based on Fine-Grained Behavioral Features", ACM Conference on Computer-Supported Cooperative Work and Social Computing (CSCW), 2021

4. Conclusion

2. Attack on Attention Check Mechanism: AC-EasyPass

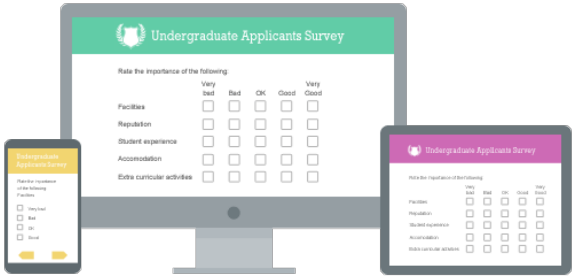
Introduction

Survey is widely used by researchers and decision makers to access vital information.

- Psychologist and sociologist: derive important studies
- Market research company: obtain feedback
- Government agency and news media: derive new policies, make important predictions

The growth and the vast accessibility of the Web have significantly facilitated the popularity of online surveys over the years.

- Potentially better targeting
- Cost saving
- Faster results
- Convenient to participants

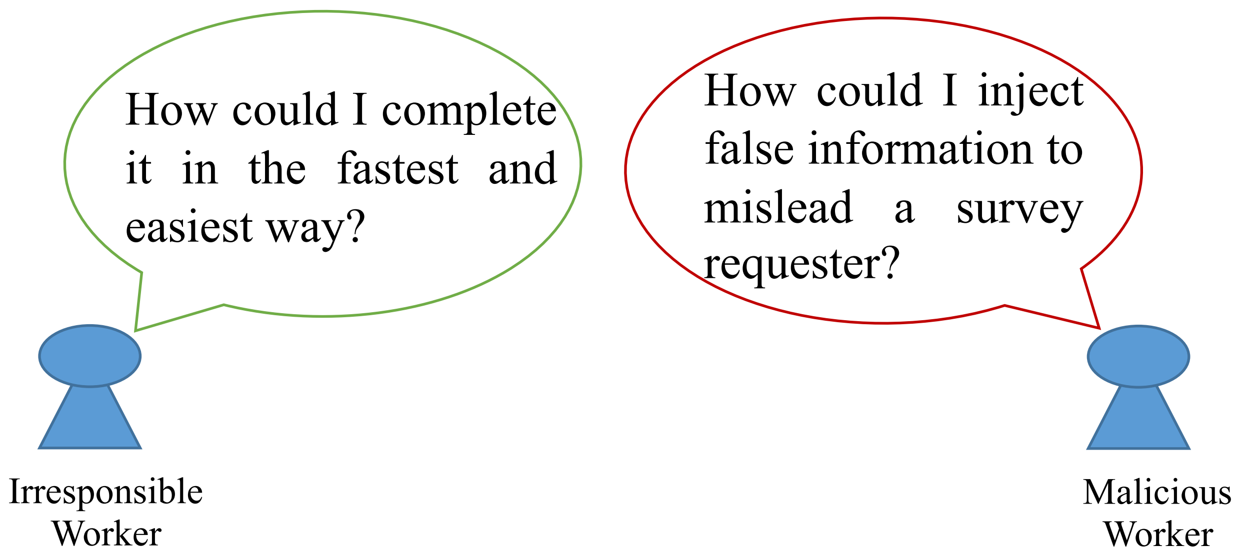


Online Survey are usually published on popular ***crowdsourcing platforms*** *such as Amazon Mechanical Turk (Mturk).*

2. Attack on Attention Check Mechanism: AC-EasyPass

Introduction: Motivation

The **quality** of survey data becomes a crucial concern for crowdsourcing service providers and researchers. Poor data quality could be caused by:



In August 2018, MTurk had a "bot" panic: psychology researchers have noticed a spike in poor quality survey responses collected on MTurk.

2. Attack on Attention Check Mechanism: AC-EasyPass

Introduction: Existing Quality Control in Online Survey

(1) Response Pattern Approach

e.g., A participant complete a survey in 5 minutes while others need 30 minutes to complete the same survey.

(2) Response Time Approach

e.g., A participant selects “neither agree nor disagree” as the response to 50 consecutive items.

Those approaches are not dependable.

(3) Attention Checking

Embeds the attention check questions that have obvious correct answers to identify inattentive respondents.

- Easy to be deployed
- Low-cost and efficient
- Appropriate for survey

2. Attack on Attention Check Mechanism: AC-EasyPass

Introduction: Two Forms of Attention Check Questions

(1) Instructional Manipulation Checks (IMCs)

Recent research on decision making shows that choices are affected by context. Differences in how people feel, their previous knowledge and experience, and their environment can affect choices. To help us understand how people make decisions, we are interested in information about you. Specifically, we are interested in whether you actually take the time to read the directions; if not, some results may tell us very much about decision making in the real world. To show that you have read the instructions, please ignore the question below about how you are feeling and instead check only the none of the above option as your answer. Thank you very much.

Please check all words that describe how you are currently feeling.
A. Excited B. Afraid C. Scared D. None of the above

(2) Instructed-response Items

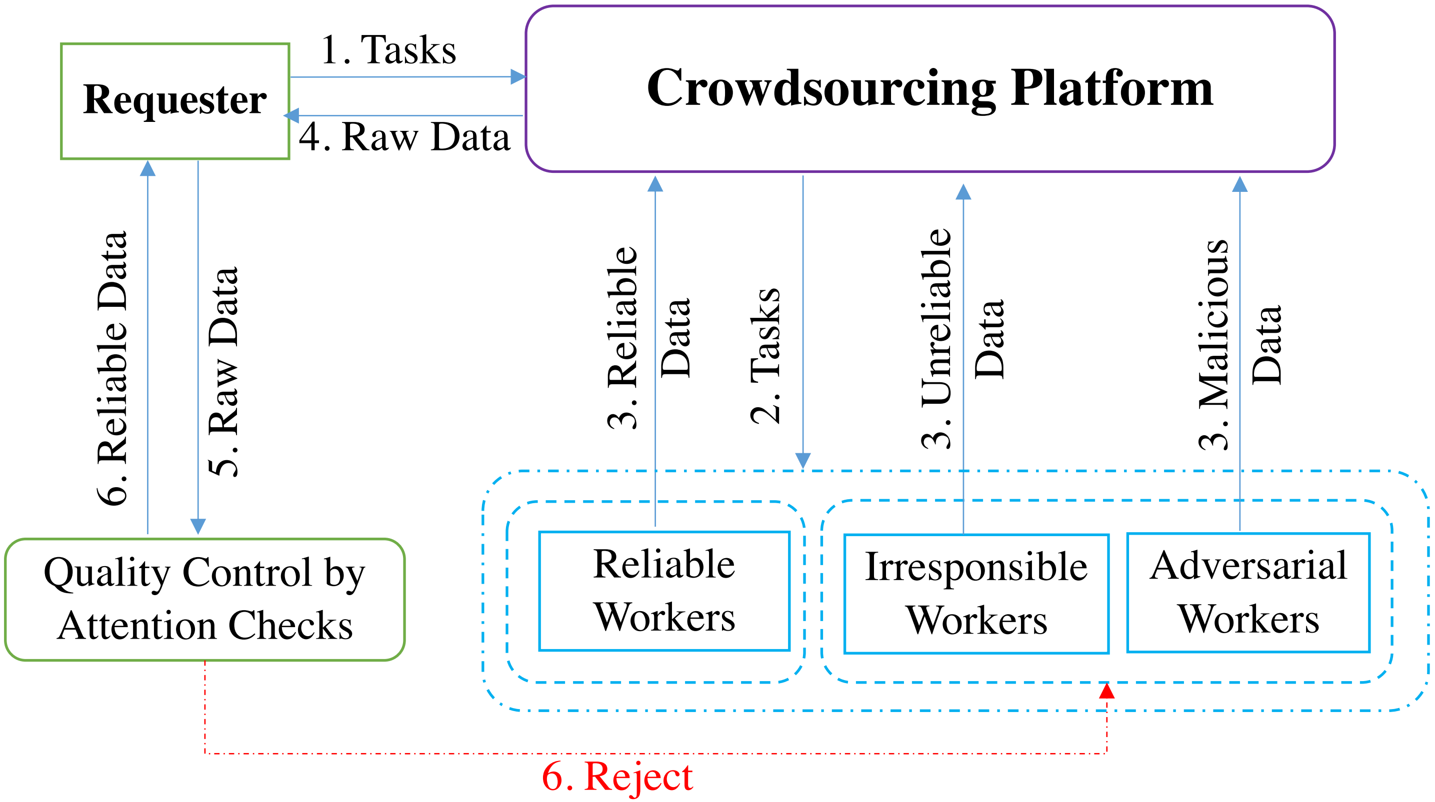
We want to test your attention, so please click on the answer Agree.

A. Disagree B. Neutral C. Agree D. Strongly agree

Is it possible for attackers to compromise the attention checking mechanism?

2. Attack on Attention Check Mechanism: AC-EasyPass

Threat Model



Vulnerability

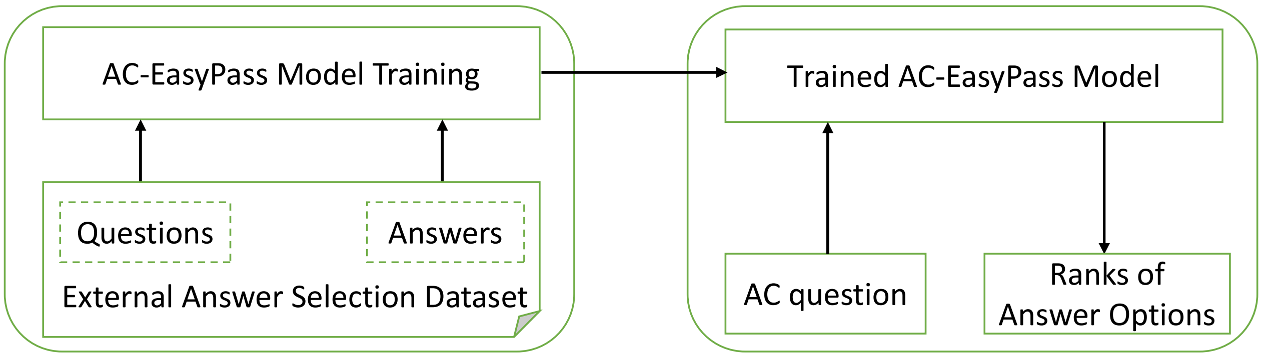
If irresponsible workers and adversarial workers could pass attention checks automatically, the quality control fails to identify poor quality data.

2. Attack on Attention Check Mechanism: AC-EasyPass

Our Approach

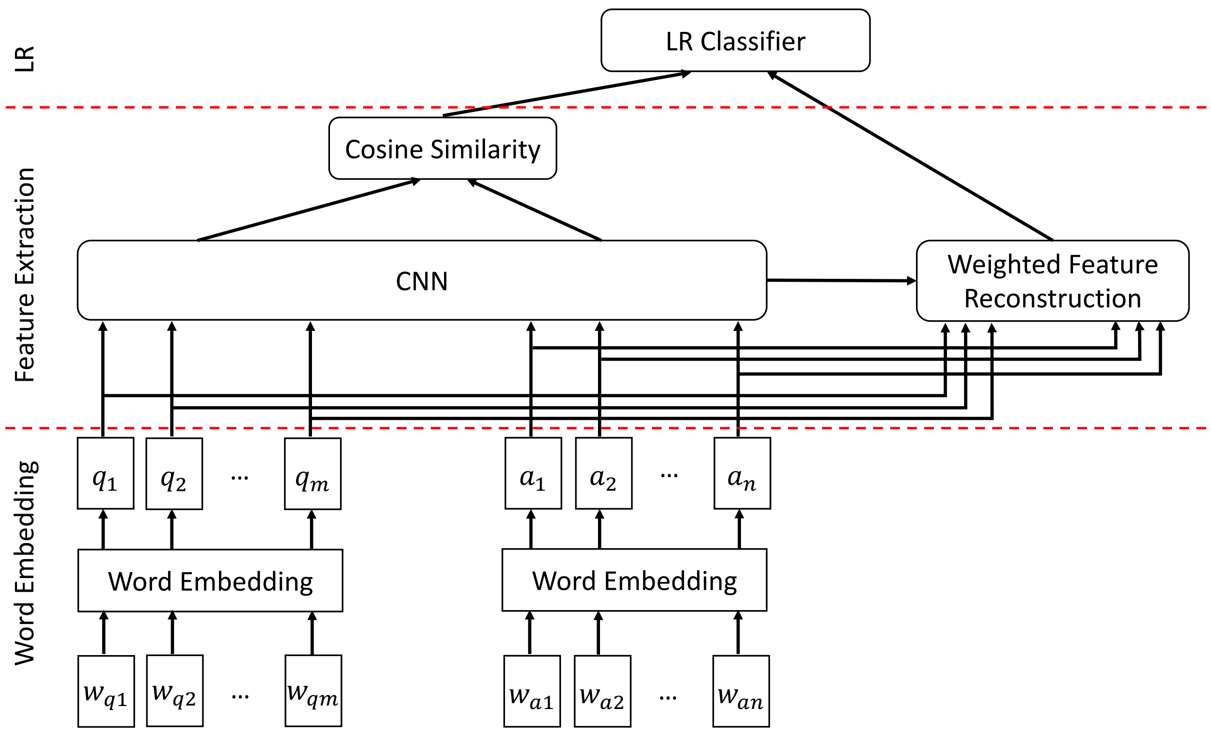
We assume that we are attackers …

- ❖ We focus on attention check questions that provide multiple choices.
- ❖ We aim to automatically analyze an attention check question and derive the correct answer. *Answer Selection*



2. Attack on Attention Check Mechanism: AC-EasyPass

AC-EasyPass Model



- ❖ Word Embedding Layer
- ❖ Feature Extraction Layer: CNN, Weighted Feature Reconstruction
- ❖ Logistic Regression (LR) Classifier Layer

2. Attack on Attention Check Mechanism: AC-EasyPass

AC-EasyPass Model

(1) Word Embedding Layer

Question: $(w_1, w_2, \dots, w_m) \rightarrow Q = (q_1, q_2, q_3, \dots, q_m) \in R^{d_0 \times m}$

Candidate Answer: $(w_1, w_2, \dots, w_n) \rightarrow A = (a_1, a_2, a_3, \dots, a_n) \in R^{d_0 \times n}$

(2) Feature Extractor Layer

- **Extract features from Convolutional Neural Network (CNN)**
  a) w-average pooling: model phrase representation
  b) all-average pooling: model sentence representation

- **Weighted Feature Reconstruction**
  a) Distance-based attention matrix $M \in R^{m \times n}$: $M_{ij} = \frac{1}{1+\|q_i - a_j\|}$
  b) Reconstruct Q and A: $\begin{cases} Q' = Af(M^T) \\ A' = Qf(M) \end{cases}$

(3) Logistic Regression (LR) Classifier Layer

All features → Logistic Regression Classifier

All the candidate answers will be ranked based on their probability to be the correct answer.

2. Attack on Attention Check Mechanism: AC-EasyPass

Setup of the Experiments

(1) Training Dataset

WikiQA: open-domain question selection dataset, including 2118 questions.

(2) Testing Dataset

Datasets	# of questions	Brief description
AC-Original	115	Collected from real-world surveys
Ans-Augmented	442	Constructed by using answer-based augmentation
Ques-Augmented	424	Constructed by using question-based augmentation

(3) Metrics for Evaluation

- Mean average precision (MAP)
- Mean reciprocal rank (MRR)
- Accuracy

(4) Reference Methods for Comparison

- *Baseline_fixed* method: simply select the first option for all questions.
- *Baseline_rand* method: simply select a random option for each question.
- *BCNN* method [2]: CNN model for modeling sentence pairs.

2. Attack on Attention Check Mechanism: AC-EasyPass

Evaluation: Effectiveness of AC-EasyPass

(1) Overall Results and Analysis

Table 1: AC-EasyPass Evaluation Results on Three Datasets.

	AC-Original			Ans-Augmented			Ques-Augmented		
Method	MAP	MRR	Accuracy	MAP	MRR	Accuracy	MAP	MRR	Accuracy
Baseline_fixed	0.3851	0.3877	0.1391	0.3979	0.4016	0.1719	0.3787	0.3861	0.1439
Baseline_rand	0.4231	0.4264	0.2043	0.3960	0.3978	0.1672	0.4146	0.4212	0.1995
BCNN	0.7889	0.7901	0.6609	0.7262	0.7270	0.5837	0.8078	0.8101	0.7028
AC-EasyPass	**0.8442**	**0.8483**	**0.7565**	**0.7969**	**0.7987**	**0.6810**	**0.8603**	**0.8661**	**0.7854**

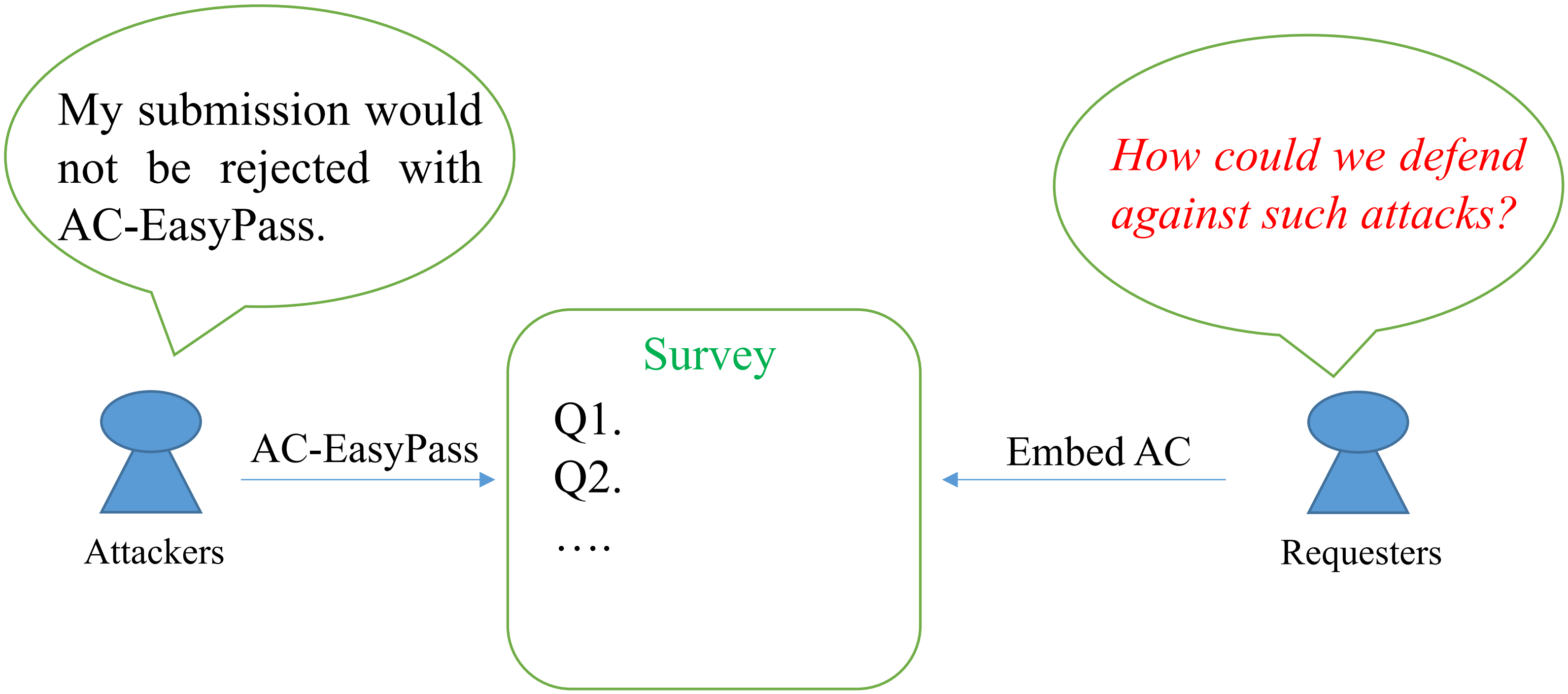
AC-EasyPass model ***outperforms*** other methods.

- AC-EasyPass model achieves 75.65% accuracy on the AC-Original dataset.

2. Attack on Attention Check Mechanism: AC-EasyPass

Defense

Now, we assume that we are defenders …



2. Attack on Attention Check Mechanism: AC-EasyPass

Defense Against AC-EasyPass

1. Adding Adversarial Phrases or Sentences

Add perturbations such as some words, phrases or sentences as noises to ***distract*** the proposed AC-EasyPass.

Three rules:

- Added perturbations should not be perceptible as irrelevant information
- Added perturbations would not change the correct answer
- Added perturbations would likely fool AC-EasyPass to select an incorrect answer.

Please click on one of options such as Disagree. We want to test your attention, so please click on the answer Agree.

A. Disagree B. Neutral C. Agree D. Strongly agree

2. Attack on Attention Check Mechanism: AC-EasyPass

Defense Against AC-EasyPass

2. Adding Typos

Use typos to ***fool*** machine comprehension models.

Two steps:

- Find the keywords
- Replace one letter with a similar or random character.

We first define high-priority letters which have similar characters.

Table 2: Some High-priority Letters and their Replacements.

Original Letter	Similar Character	Replacement Example
q	9	question → 9uestion
o	0	other → 0ther
z	2	zero → 2ero
l	1	select → se1ect
u	v	true → trve
s	5	classified → cla5sified

2. Attack on Attention Check Mechanism: AC-EasyPass

Evaluation of the Two Defense Methods

(1) Overall Results and Analysis

AC-Original + Adding Adversarial Sentences/Phrases → **AC-Original-Adversarial**
AC-Original + Adding Typos → **AC-Original-Typos**

Table 3: Effectiveness of the Two Defense Methods on Decreasing AC-EasyPass Performance.

Dataset	MAP	MRR	Accuracy
AC-Original	0.8442	0.8483	0.7565
AC-Original-Adversarial	0.7144	0.7178	0.5478
AC-Original-Typos	0.5247	0.5326	0.2957

Both methods can ***to some extent decrease*** the accuracy of our AC-EasyPass attacks.

- Adding adversarial sentences contributes to an over *10% decrease* in both MAP and MRR
- Adding typos leads to a more than *30% decrease* in both MAP and MRR.

2. Attack on Attention Check Mechanism: AC-EasyPass

Limitations of the Two Defense Methods

(1) Adding Adversarial Phrases or Sentences

This defense method will become less effective if attackers include some adversarial sentences to train AC-EasyPass and improve its robustness.

Ques-Augmented-Adversarial dataset: Apply adding adversarial sentences method to the Ques-Augmented dataset

- **Adversarial Training**: 0.75 MAP, 0.76 MRR, and 0.61 accuracy on the AC-Original-Adversarial dataset.

(2) Adding Typos

Attackers can leverage spelling check techniques to correct those typos and improve the robustness of AC- EasyPass.

- **Spelling Check Service** (Microsoft Azure): 59.7% of the questions in the AC-Original-Typos dataset can be completely corrected, while 8.4% of the questions can be partially corrected.

Both defense methods are fragile and defense remains a challenging task.

2. Attack on Attention Check Mechanism: AC-EasyPass

Summary

(1) We performed the first study to investigate the vulnerabilities of the attention check mechanism.

(2) We proposed and designed AC-EasyPass, an attack framework to easily pass attention check questions.

(3) We constructed the first attention check question dataset that consists of both original and augmented questions, and demonstrated that AC-EasyPass is effective on those questions.

(4) We also explored two simple defense methods, adding adversarial sentences and adding typos, for survey designers to mitigate the risks posed by AC-EasyPass.

Outline

1. Introduction of Crowdsourcing and Data Quality
2. Attack on Attention Check Mechanism: AC-EasyPass
   - "Attention Please: Your Attention Check Questions in Survey Studies Can Be Automatically Answered", The Web Conference (WWW), 2020
3. Fine-grained Behavior-based Quality Control (FBQC)
   - "Quality Control in Crowdsourcing based on Fine-Grained Behavioral Features", ACM Conference on Computer-Supported Cooperative Work and Social Computing (CSCW), 2021
4. Conclusion

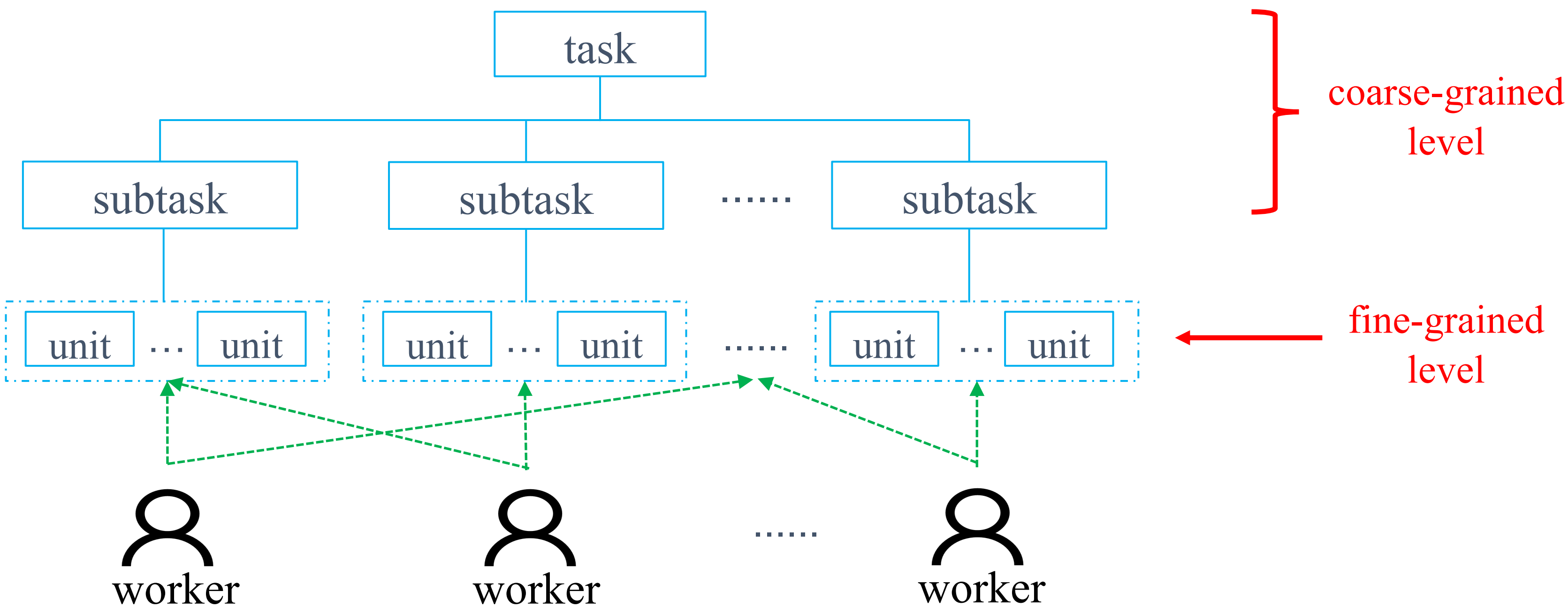
3. Fine-grained Behavior-based Quality Control (FBQC)

Introduction

Crowdsourcing is popular for large-scale data collection.

- Job requesters break a large *task* into many smaller *subtasks*, each of which consists of one or more annotation *units*.

- Annotations collected from workers could be divided into different levels of granularity: ***unit level***, ***subtask level***, and ***task level***.



3. Fine-grained Behavior-based Quality Control (FBQC)

Introduction

Textual Emotion Recognition

Multiple dialogues by a worker → task level

subtask level

Chandler: Good job Joe! Well done! Top notch! (required)	○ Anger ○ Sadness ○ Joy ○ Fear ○ Disgust ○ Surprise ○ Neutral
Joey: You liked it? You really liked it? (required)	○ Anger ○ Sadness ○ Joy ○ Fear ○ Disgust ○ Surprise ○ Neutral
Chandler: Oh-ho-ho, yeah! (required)	○ Anger ○ Sadness ○ Joy ○ Fear ○ Disgust ○ Surprise ○ Neutral
Joey: Which part exactly? (required)	○ Anger ○ Sadness ○ Joy ○ Fear ○ Disgust ○ Surprise ○ Neutral
Chandler: The whole thing! Can we go? (required)	○ Anger ○ Sadness ○ Joy ○ Fear ○ Disgust ○ Surprise ○ Neutral
Joey: Oh no-no-no, give me some specifics. (required)	○ Anger ○ Sadness ○ Joy ○ Fear ○ Disgust ○ Surprise ○ Neutral
Chandler: I love the specifics, the specifics were the best part! (required)	○ Anger ○ Sadness ○ Joy ○ Fear ○ Disgust ○ Surprise ○ Neutral
Joey: Hey, what about the scene with the kangaroo? Did-did you like that part? (required)	○ Anger ○ Sadness ○ Joy ○ Fear ○ Disgust ○ Surprise ○ Neutral
Chandler: I was surprised to see a kangaroo in a World War I epic. (required)	○ Anger ○ Sadness ○ Joy ○ Fear ○ Disgust ○ Surprise ○ Neutral

unit level

3. Fine-grained Behavior-based Quality Control (FBQC)

Introduction: Existing Quality Control in Crowdsourcing

(1) Gold Standard

Compare a worker's submissions against a set of labeled high-quality data

(2) Redundancy

Assign the same subtask to a number of workers and then infer the consensus label by using aggregation, such as Majority Voting

(3) Behavior Analysis

Estimate the quality of submissions by analyzing workers' behavioral data (e.g., mouse clicks and keypresses)

Quality control in crowdsourcing is critical and challenging.

3. Fine-grained Behavior-based Quality Control (FBQC)

Introduction: Limitations of Existing Behavior Analysis

(1) Mainly focused on coarse-grained behaviors

Coarse-grained behavioral analysis can lead to the inclusion of low-quality data, exclusion of high-quality data, and/or manipulation by malicious workers

(2) Do not consider subtasks consisting of varying number of units

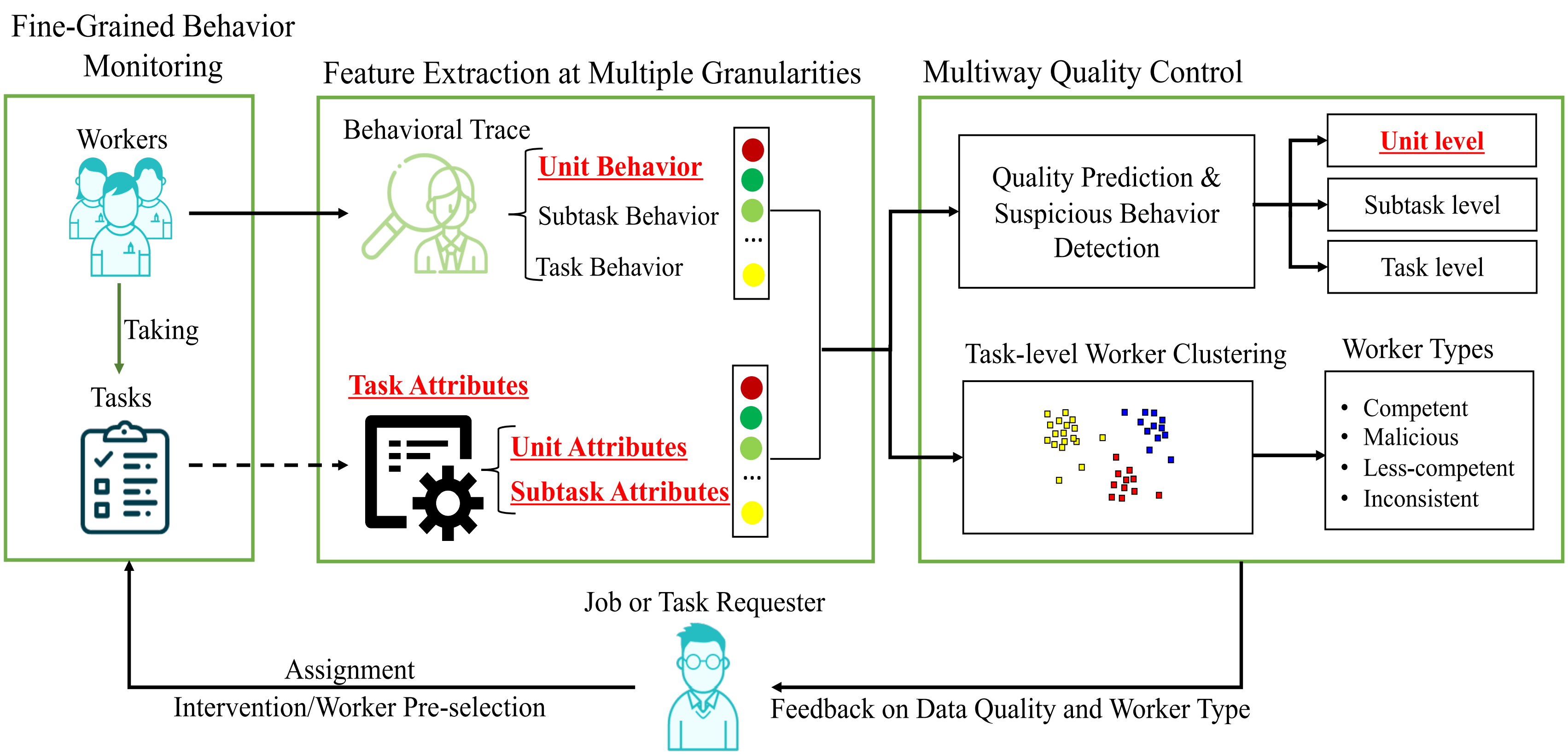
(3) Lack of behavior analysis for subjective tasks

Research Goal

Investigate feasibility and benefits of using fine-grained behavioral features for quality control in crowdsourcing.

3. Fine-grained Behavior-based Quality Control (FBQC)

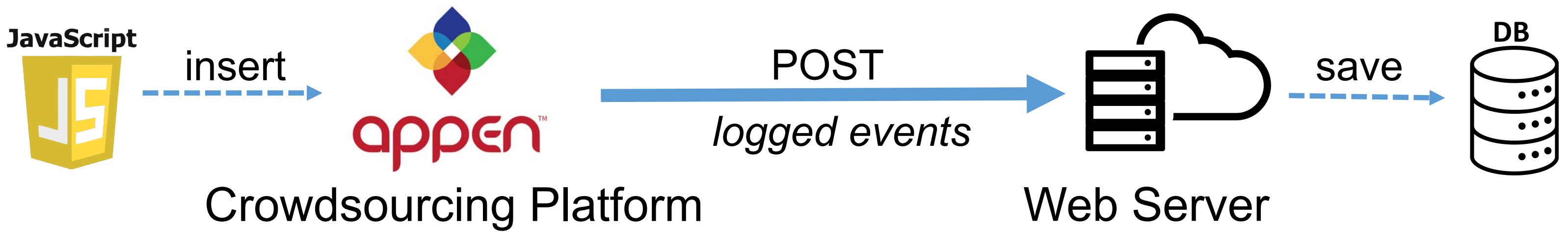
Proposed FBQC Framework



3. Fine-grained Behavior-based Quality Control (FBQC)

Components of FBQC Framework

(1) Fine-Grained Behavior Monitoring



(2) Feature Extraction at Multiple Granularities

- **Behavioral Trace**
  Unit Behavioral (UB) Features, e.g., time spent on a unit.
  subTask Behavioral (TB) Features, e.g., total time spent on a subtask.

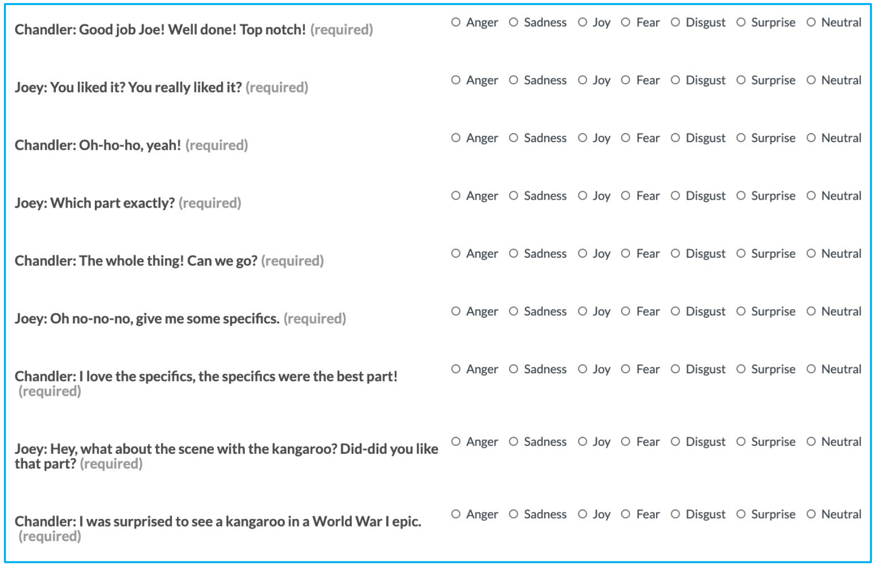
- **Task Attributes**
  Unit Attribute (TA) Features, e.g., the length/size of a unit.
  subTask Attribute (TA) Features, e.g., the number of units in a subtask.

3. Fine-grained Behavior-based Quality Control (FBQC)

Components of FBQC Framework

(2) Feature Extraction at Multiple Granularities

(a) Image Task



(b) Text Task

Image Task (Visual Object Detection)
Unit Level: Each bounding box
subTask Level: Each image
Task Level: All images completed by a worker

Text Task (Textual Emotion Recognition)
Unit Level: Each utterance
subTask Level: Each dialogue
Task Level: All dialogues completed by a worker

3. Fine-grained Behavior-based Quality Control (FBQC)

Components of FBQC Framework

(2) Feature Extraction at Multiple Granularities

Table 3. Fine-Grained Features Extracted for Each Unit

Feature Type	Feature Name	Description
Unit Behavioral (UB) Features	time_on_unit	Time spent on a unit task, i.e., a bounding box in the image task or an utterance in the text task.
	total_[X]_events	The number of logged events of type X for a bounding box or an utterance where X could be one in {create, remove} in the image task, or {clicks, keypresses, checks} in the text task.
	num_change_annotation	The number of times that a worker deletes a bounding box or changes an option.
	events_around_annotation	The number of logged events immediately around the annotation action for a unit, including clicks, keypresses, movements, etc.
	movement_speed_unit	The mean, median, and standard deviation of mouse movement speed within the created bounding box or the utterance.
	speed_around_annotation	The mean, median, and standard deviation of mouse movement speed before/after creating a bounding box or selecting an option for an utterance.
Unit Attribute (UA) Features	unit_attributes	The attributes of a unit, i.e., the size and entropy of a bounding box in the image task, or the number of words and prepositions of an utterance in the text task.

Table 4. Coarse-Grained Features Extracted for Each Subtask

Feature Type	Feature Name	Description
subTask Behavioral (TB) Features	time_on_subtask	Total time spent on a subtask, i.e., an image or a dialogue.
	total_[X]_events	The number of logged events of type X for a subtask where X could be one in {create, remove} in the image task, or {clicks, keypresses, checks} in the text task.
	time_on_instruction	Time spent by a worker on reading the task instruction before starting the first unit.
	tBeforeInput	Time taken by a worker before creating the first bounding box or choosing the first option in a subtask.
subTask Attribute (TA) Features	subtask_attributes	The attributes of a subtask, i.e., the size and entropy of an image in the image task, or the number of utterances of a dialogue in the text task.

3. Fine-grained Behavior-based Quality Control (FBQC)

Components of FBQC Framework

(3) Multiway Quality Control

- **Quality prediction for objective tasks**
  Objective tasks have ground-truths, e.g., object detection
  Train supervised models based on extracted features to predict data quality.

- **Suspicious behavior detection for subjective tasks**
  Subjective tasks do not have ground-truths, e.g., emotion recognition, surveys.
  Define a set of rules to identify suspicious behaviors.

- **Unsupervised worker categorization**
  Apply a clustering algorithm (K-Means) to group workers.

3. Fine-grained Behavior-based Quality Control (FBQC)

Task Design

(1) Image Task (Visual Object Detection)

Dataset: 200 sampled images from the Open Image dataset
Number of units per subtask: 3~10
Number of workers for each subtask: 10

(2) Text Task (Textual Emotion Recognition)

Dataset: 420 sampled dialogues from the MELD dataset
Number of units per subtask: 4~24
Number of workers for each subtask: 10

Table 2. Summary of the Collected Data including the Number of Completed Units and Subtasks.

Task Type	# images or dialogues	# units	# subtasks	# workers
Image	200	10,984	1,948	258
Text	460	49,395	4,451	427

3. Fine-grained Behavior-based Quality Control (FBQC)

Evaluation 1: Quality Prediction (objective tasks)

Objective Task: Visual Objective Detection

Method: leverage fine-grained features to build supervised machine learning models

(1) Unit level quality prediction

Table 5. Unit Level Quality Prediction (UB - Unit Behavioral Features, UA - Unit Attribute Features)

Model Type	Baseline	SVR/SVC			RFR/RFC		
Features	-	UB	UA	UB&UA	UB	UA	UB&UA
Regression (MSE×100)	2.58	2.29	2.86	2.23	1.94	2.15	**1.84**
Classification (Accuracy)	60.8%	69.6%	63.6%	69.9%	69.5%	61.2%	**70.0%**

(2) Subtask level quality prediction

Table 7. Subtask Level Quality Prediction (TB - subTask Behavioral Features, UB - Unit Behavioral Features, TA - subTask Attribute Features. Here UB∗ features are statistical features derived from UB features of all units in a subtask.)

Model Type	Baseline	DT-AF [8, 34]	RF-AF [8]	RF-SF [8]	RFR/RFC				
Features	-	-	-	-	TB	UB*	TB&UB*	TA	TB&UB*&TA
Regression (MSE×100)	4.16	3.1	1.8	1.5	1.57	0.89	0.90	1.07	**0.76**
Classification (Accuracy)	66.3%	65.4%	74.0%	73.9%	67.8%	83.1%	82.7%	75.8%	**83.4%**

3. Fine-grained Behavior-based Quality Control (FBQC)

Evaluation 1: Quality Prediction (objective tasks)

(3) Task level quality prediction

Table 8. Task Level Quality Prediction (TB - subTask Behavioral Features, UB - Unit Behavioral Features. Here TB# features are statistical features derived from TB features of all subtasks in a task, and UB# features are statistical features derived from UB features of all units in a task)

Model Type	Baseline	Gold Standard	SVR/SVC			RFR/RFC		
Features	-	-	$TB^{\#}$	$UB^{\#}$	$TB^{\#}$&$UB^{\#}$	$TB^{\#}$	$UB^{\#}$	$TB^{\#}$&$UB^{\#}$
Regression (MSE×100)	3.44	1.60	1.58	1.41	1.53	1.26	0.89	**0.90**
Classification (Accuracy)	72.0%	74.6%	78.5%	83.1%	82.4%	81.2%	82.0%	**84.7%**

3. Fine-grained Behavior-based Quality Control (FBQC)

Evaluation 2: Suspicious Behavior Detection (subjective tasks)

Subjective Task: Textual Emotion Recognition

Method: design rules for detecting suspicious behaviors

Rules:

1) the time spent on a unit (time on unit) is less than a threshold tr;
2) there is no mouse click or keypress observed in a unit;
3) none of radio buttons in a unit has been put on focus during the subtask execution.

(1) Overall performance of 200 sampled utterances

Table 10. Performance of Fine-Grained Level Suspicious Behavior Detection

(a) Confusion Matrix.

		Manual Inspection	
		Suspi.	Non-Suspi.
Automated Detection	Suspi.	186	14
	Non-Suspi.	26	174

(b) Overall Performance.

Accuracy	Precision	Recall	F1 score
90.0%	93.0%	87.7%	90.3%

3. Fine-grained Behavior-based Quality Control (FBQC)

Evaluation 3: Unsupervised Worker Categorization

Tasks: Visual Objective Detection & Textual Emotion Recognition

Worker Types

(1) Competent Workers: provide high-quality submissions for all their subtasks.
(2) Malicious Workers: be purely money-driven, and attempt to compete each subtask with the least time or effort.
(3) Less-competent Workers: complete all subtasks successfully with sufficient time but provide low-quality data.
(4) Inconsistent Workers: act like a competent or less-competent worker in some subtasks while act like a malicious worker in others.

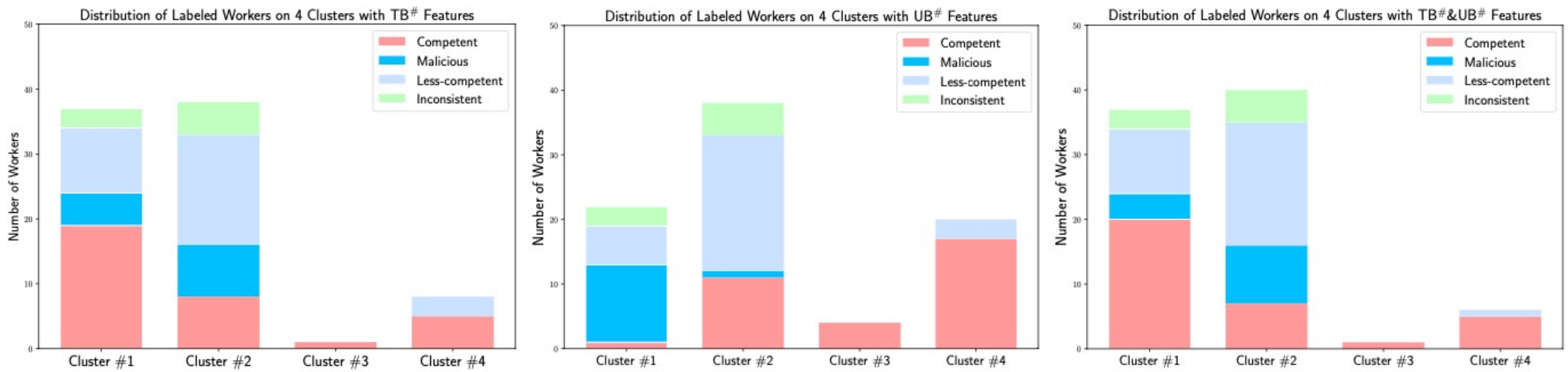
Table 11. Manually Identified or Labeled Types for Sampled Workers

Task (# workers)	Competent	Malicious	Less-competent	Inconsistent
Image (85)	34	13	30	8
Text (103)	39	38	20	6

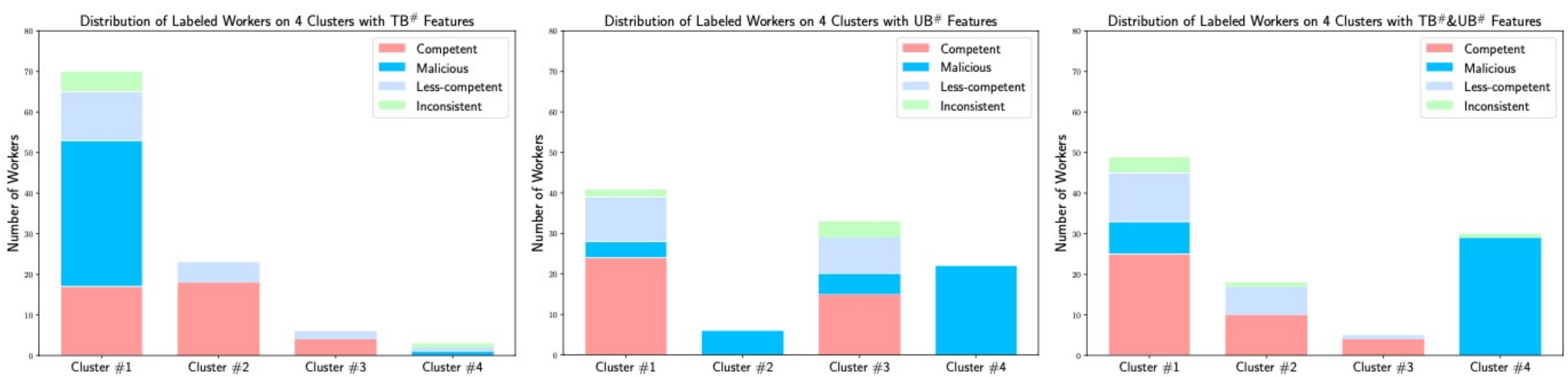
3. Fine-grained Behavior-based Quality Control (FBQC)

Evaluation 3: Unsupervised Worker Categorization

Figure 7. Distribution of Manually Labeled Workers on Four Clusters in Six Different Experiments



(a) TB# features (Image Task) (b) UB# features (Image Task) (c) TB#&UB# features (Image Task)



(d) TB# features (Text Task) (e) UB# features (Text Task) (f) TB#&UB# features (Text Task)

3. Fine-grained Behavior-based Quality Control (FBQC)

Discussion

Generalizability, Deployability, and Scalability of the FBQC Framework

Table 12. Examples of Other Important Crowdsourcing Tasks and their Data at Different Granularities.

Crowdsourcing Task	**Unit Data**	**Subtask Data**	**Task Data**
Image Segmentation [11]	The outline of a target object provided by a worker for an image on a webpage.	All outlines of target objects provided by a worker for all images on a webpage.	All outlines of target objects provided by a worker in the entire image segmentation task.
Image Transcription [6]	The content in a text input field provided by a worker for an image on a webpage.	All contents in text input fields provided by a worker for all images on a webpage.	All contents in text input fields provided by a worker in the entire image transcription task.
Text Annotation by Token [23]	The token selected by a worker for a target class in a paragraph on a webpage.	All tokens selected by a worker for target classes in all paragraphs on a webpage.	All tokens selected by a worker for target classes in the entire text annotation task.
Reading Comprehension [34]	The answer provided by a worker to a question in a paragraph on a webpage.	All answers provided by a worker to questions in all paragraphs on a webpage.	All answers provided by a worker to questions in the entire reading comprehension task.
Survey [26, 30]	The response provided by a worker to a question in a survey.	All responses provided by a worker to a section or webpage of questions in a survey.	All response provided by a worker to all questions in the entire survey.
Relevance Judgment [8, 18]	The relevant score provided by a worker for a query-document pair on a webpage.	All relevant scores provided by a worker for all query-document pairs on a webpage.	All relevant scores provided by a worker for all query-document pairs in the entire relevance judgment task.

3. Fine-grained Behavior-based Quality Control (FBQC)

Discussion

Table 14. Coarse-Grained Features for Each Subtask in Other Important Crowdsourcing Tasks

Feature Type	Feature Name	Description
subTask Behavioral (TB) Features	time_on_subtask	Total time spent on a subtask, i.e., the subtask data (Column 3 of Table 12) for certain crowdsourcing task shown in Table 12.
	total_[X]_events	The number of logged events of type X on a webpage where X could be one in {clicks, keypresses, checks, ...} for certain crowdsourcing task.
	time_on_instruction	Time spent by a worker on reading the task instruction before starting the first unit in a subtask.
	tBeforeInput	Time taken by a worker before creating the first annotation (i.e., the unit data) in a subtask.
subTask Attribute (TA) Features	subtask_attributes	The attributes of a subtask (Column 3 of Table 12), e.g., (1) the size, entropy and image gradients of all given images on a webpage in the Image Segmentation task, (2) the size and entropy of all given images on a webpage in the Image Transcription task, (3) the number of tokens in a paragraph in the Text Annotation by Token task, (4) the number of sentences/words of given paragraphs, and the number of questions in the Reading Comprehension task, (5) the number of questions in the Survey task, and (6) the number of query-document pairs on a webpage in the Relevance Judgment task.

(UA) Features	unit_attributes	(4) the number of sentences/words of given paragraphs, the co-occurrence words between paragraphs and a question in the Reading Comprehension task, (5) the number of words of a question in the Survey task, and (6) the number of query words in each document in the Relevance Judgment task.

3. Fine-grained Behavior-based Quality Control (FBQC)

Summary

(1) We explore the feasibility and benefits of using fine-grained behavioral features for quality control at the fine-grained level and also at higher levels.

(2) We designed and implemented the FBQC framework that specifically extracts fine-grained behavioral features to provide three quality control mechanisms:

- quality prediction for objective tasks
- suspicious behavior detection for subjective tasks
- unsupervised worker categorization

(3) We conducted two real-world crowdsourcing experiments and demonstrated that using fine-grained behavioral features are feasible and beneficial in all three quality control mechanisms.

Conclusion

- "Attention Please: Your Attention Check Questions in Survey Studies Can Be Automatically Answered", The Web Conference (WWW), 2020
    - Attention check questions can be automatically passed.
    - Defense methods can be fragile and defense remains a challenging task.

- "Quality Control in Crowdsourcing based on Fine-Grained Behavioral Features", ACM Conference on Computer-Supported Cooperative Work and Social Computing (CSCW), 2021
    - Coarse-grained behavior based quality control is insufficient.
    - Our proposed FBQC achieves better performance for quality control.

Quality control in crowdsourcing is important yet still challenging!

Thank you! Q&A