Data Science for Transportation (2026) 8:21
https://doi.org/10.1007/s42421-026-00158-4

RESEARCH

Explainable Data-Driven Multi-filter Framework for Bot Detection in Incentivized Online Surveys

Andrew Ruger[1] · Mohammed Abdalazeem[2] · Eleni Christofa[2] · Jimi Oke[2]

Received: 25 November 2025 / Revised: 10 April 2026 / Accepted: 21 April 2026

Abstract

Online surveys have become increasingly popular for academic research due to their cost-effectiveness and ability to reach large populations. However, the use of monetary incentives to encourage participation has led to widespread infiltration by automated bots designed to complete surveys fraudulently. This study presents a transparent, multi-filter framework to detect and remove bot responses, using an incentivized online transportation survey on cycling preferences as a case study. The survey received 12,020 responses over 94 days, with clear evidence of bot activity including 3951 submissions (32.9%) concentrated on a single day. To secure the dataset, we developed an 11-filter pipeline spanning four categories: platform risk scores, duplicate identity detection, behavioral interaction patterns, and device consistency. The multi-filter pipeline removed 94.8% of the submissions. Sensitivity analyses reveal that while platform risk scores (reCAPTCHA, fraud score) screen the bulk of automated traffic, behavioral timing and device checks are essential for catching sophisticated bots. To establish ground-truth performance, we conducted a blind expert review on a data subsample, achieving an 88.9% agreement rate between the automated pipeline and independent human judgment. Finally, to audit the internal coherence of the rules, we trained an explainable XGBoost classifier. The model independently recovered the filter decisions with an F1 score of 0.94 for the human class (precision 0.90, recall 0.98), proving the pipeline captures statistically separable behavioral clusters rather than arbitrary noise. This open-source framework provides researchers with a generalizable, evidence-based template for ensuring data integrity in online surveys.

Keywords Survey bots · Data quality · Fraud detection · Online surveys · reCAPTCHA · Multi-filter pipeline · SHapley Additive exPlanations (SHAP) · Explainable machine learning

Introduction

The rapid expansion of online survey platforms has transformed behavioral research by providing cost-effective access to large, geographically distributed populations (Ponto 2015; Regmi et al. 2016). Online data collection enables researchers to reach substantially more participants than traditional methods while offering speed and convenience in data collection (Singh and Sagar 2021). Transportation research increasingly uses online surveys to investigate cycling behavior, infrastructure preferences, and travel patterns (Forsyth 2010; Fang 2020; Hardinghaus and Papantoniou 2020). Since these surveys often rely on stated-preference tools to build behavioral models and inform policy, automated or disengaged responses can severely distort the resulting conclusions (Petrik et al. 2016). Furthermore, these methods usually require repeated tasks and sustained attention, making inter-

Andrew Ruger and Mohammed Abdalazeem contributed equally to this work.

✉ Mohammed Abdalazeem
mamohammed@umass.edu

Andrew Ruger
rugerand@grinnell.edu

Eleni Christofa
echristofa@umass.edu

Jimi Oke
jimi@umass.edu

[1] Department of Economics, Grinnell College, 1226 Park Street, Grinnell, IA 50112, USA

[2] Department of Civil and Environmental Engineering, University of Massachusetts Amherst, 130 Natural Resources Rd, Amherst, MA 01003, USA

Published online: 13 June 2026

action signals and completion times highly valuable for filtering out bad data after collection (Conrad et al. 2017).

However, the combination of online accessibility and financial incentives has created significant challenges for data integrity. The anonymity inherent in online surveys, coupled with monetary rewards, renders them vulnerable to fraudulent responses generated by software programs known as survey bots (Pozzar et al. 2020; Storozuk et al. 2020; Yarrish et al. 2019; Godinho et al. 2020). These bots can complete surveys autonomously, generate random or templated responses, and rapidly exhaust research budgets while contaminating data sets with invalid data (Griffin et al. 2022; Xu et al. 2022; Buchanan and Scofield 2018). This can result in misguided strategies and resource misallocation (Caven et al. 2025).

The scale of this problem has grown substantially in recent years. Studies in various disciplines report fraudulent participant contamination rates ranging from 20% to more than 80% of survey responses, even when preventive measures are employed (Agans et al. 2024; Thompson and Utz 2025). This trend threatens the validity of research findings and strains resources that could otherwise support legitimate participants and strengthen research findings.

Despite the severity of this threat, there remains a lack of practical tools and generalizable approaches for detecting and filtering bot responses (Griffin et al. 2022; Kennedy et al. 2020). Existing methods often rely on single detection techniques or platform defaults that may not be optimal for specific research contexts (Teitcher et al. 2015; Dennis et al. 2020). Commercial solutions frequently operate as black boxes without transparency in their detection logic, while research-based approaches have struggled to translate from controlled simulations to real-world applications (Irish and Saba 2023). Furthermore, there is a dearth of evidence-based guidance for setting detection thresholds and systematically combining multiple indicators.

To address these gaps, we present a systematic framework for bot detection and filtering, using a large-scale survey about cycling preferences and schematic maps as a case study. Our transparent framework addresses the critical need for replicable and adaptable bot-detection methods in incentivized online research. While prior studies have utilized multiple screening criteria, this study advances the methodology by integrating the following components into a complete and reusable workflow. First, we provide a fully parameterized, end-to-end multi-stage filtering pipeline that integrates platform-provided fraud-detection tools with behavioral indicators and device consistency checks, with explicit thresholds that can be tuned to different instruments. Second, we document robustness testing through threshold sensitivity analyses and ablation studies to quantify how retention changes when criteria are varied or removed. Third, we audit the internal coherence of the rule-based multi-filter pipeline by training a supervised learning model that independently reproduces the filter-based decisions with an F1 score of 0.94 and provides feature-level explanations of filter impacts. The implementation is released as open-source code. Overall, this study lays a foundation for transparent and transferable data-driven filtering approaches that will be critical to strengthening the integrity of online survey data in the information and artificial intelligence (AI) age. While our case study is a transportation survey, we present the framework as a general workflow whose thresholds should be calibrated to the specific survey design and incentive structure.

Background and Relevant Work

Evolution of Bot-Detection Technologies

Over the past 2 decades, online (internet-based) survey platforms, distributed via desktops, tablets, and smartphones, have revolutionized how researchers collect data from survey participants. SurveyMonkey, founded in 1999, and Qualtrics, launched in 2002, emerged as dominant platforms that made internet-based survey deployment more accessible to researchers (Fielding et al. 2017). Newer entrants like Typeform have joined these established platforms, offering conversational survey interfaces, while Google Forms and Microsoft Forms provide free alternatives for basic data collection needs (Vasantha Raju 2016). These platforms enable researchers to reach substantially more participants than traditional in-person or telephone surveys, with financial or other incentives commonly used to encourage participation and increase sample sizes (Storozuk et al. 2020).

However, the combination of online accessibility and monetary incentives creates an attractive target for survey bots, which are automated software programs designed to complete surveys fraudulently to obtain the offered compensation (Caven et al. 2025). These bots not only compromise data quality but also exhaust research budgets by claiming incentives intended for legitimate participants, which is particularly problematic for smaller organizations with limited resources (Griffin et al. 2022).

To combat these threats, survey platforms and researchers have relied on bot-detection technologies, with Google's reCAPTCHA being one of the most widely deployed solutions (Sivakorn et al. 2016; Searles et al. 2023). CAPTCHA, an acronym for "Completely Automated Public Turing test to tell Computers and Humans Apart," was designed to present challenges that are easy for humans but difficult for computers to solve. Google acquired reCAPTCHA in 2009 and has since developed multiple versions: the original text-based system, reCAPTCHA v2 with image selection challenges,

and reCAPTCHA v3 which operates invisibly using behavioral analysis (Ahn and Cathcart 2009).

Initially, reCAPTCHA demonstrated strong effectiveness, as research from the early 2010s showed that it successfully blocked most automated attempts. However, empirical evidence now demonstrates that these bot detectors are increasingly ineffective against sophisticated attacks. Despite implementing reCAPTCHA screening, 61.8% of responses to a 2020 health survey were identified as bots through post hoc analysis using duplicate IP addresses, outlier response times, and inconsistent demographic responses (Griffin et al. 2022). A 2014 Google study found that AI could solve text-based CAPTCHAs with 99.8% accuracy, while current research shows bots can solve image-based reCAPTCHAs with 70–100% accuracy (Plesner et al. 2024). Recent studies demonstrate that bots can solve 100% of reCAPTCHA v2 challenges, surpassing the 68–71% success rates reported in earlier work (Plesner et al. 2024).

Beyond reCAPTCHA, survey platforms like Qualtrics offer built-in fraud-detection features, including IP tracking, device fingerprinting, and behavioral analysis. These platforms assign risk scores to submissions and flag suspicious patterns. Yet, sophisticated bots continue to evade these measures through virtual private network (VPN) usage, browser automation tools, and machine learning algorithms trained to solve challenges (Xu et al. 2022).

Survey Design Methods for Bot Prevention

Given the limitations of automated detection technologies, researchers have developed survey design strategies to discourage bot participation and facilitate post hoc identification. These preventative measures aim to make surveys less attractive or more difficult for bots to complete successfully.

Question design approaches include incorporating attention check questions, open-ended responses that require contextual understanding, and intentionally incorrect options in multiple-choice questions that bots might select (Caven et al. 2025). Open-ended questions have historically discouraged older rule-based bots. However, modern language model-based bots can generate human-like responses, diminishing the effectiveness of this method (Höhne et al. 2025). Researchers can also implement eligibility screening questionnaires before granting survey access, although determined attackers can program bots to pass common screening criteria (Agans et al. 2024).

Financial incentive structures significantly influence bot attraction. Switching from guaranteed per-participant payments to lottery-based systems with larger randomly allocated prizes can reduce bot submissions without changing total compensation budgets, as the uncertain payoff discourages automated attempts (Griffin et al. 2022). However, this approach may also reduce legitimate participation from individuals who prefer guaranteed compensation.

Platform security features offer additional protection, albeit with limitations. Qualtrics security settings can prevent multiple submissions from the same device and exclude survey links from search engine indexes, reducing discoverability by automated crawlers (Xu et al. 2022). Yet these measures fail against sophisticated fraudulent actors who rely on VPNs or residential proxy networks to rotate IP addresses and on browser automation frameworks that simulate human-like interaction patterns, allowing them to evade IP-, cookie-, and behavior-based defenses (Plesner et al. 2024; Zhang et al. 2022). Moreover, overly restrictive security settings can inadvertently block legitimate participants from accessing surveys on shared networks or in institutional settings.

Detection Frameworks and Statistical Approaches

While single-indicator detection methods provide some protection, comprehensive frameworks that combine multiple indicators show greater promise. Post-submission analysis remains crucial for identifying bot responses that evade initial screening.

Traditional filtering approaches examine multiple data quality indicators. IP address duplication can warrant immediate exclusion (Thompson and Utz 2025), though shared institutional networks can produce false positives. Completion time analysis flags surveys finished unusually fast or slow relative to human benchmarks (Xu et al. 2022; Hardesty et al. 2024). Temporal clustering identifies bot attacks when multiple submissions occur within seconds, while submissions during unusual hours (12 AM to 6 AM) raise suspicion (Storozuk et al. 2020; Caven et al. 2025; Agans et al. 2024). Platform-generated scores from Qualtrics and reCAPTCHA enable threshold-based filtering. However, optimal cutoff values remain unclear (Xu et al. 2022; Thompson and Utz 2025).

Other approaches recognize that few individual indicators definitively prove bot activity. Researchers increasingly adopt multi-criteria frameworks where responses must trigger multiple warning signs before exclusion (Thompson and Utz 2025). However, existing literature provides limited guidance on which combinations of flags optimize detection accuracy while minimizing false positives.

Advanced statistical methods show promise for bot detection but face translation challenges from controlled environments to real-world applications. Methods that use metrics such as Mahalanobis distance and person-total correlation demonstrate effectiveness in simulated conditions, but complexities arise in practical application where the true number of bot responses remains unknown (Dupuis et al. 2019; Irish and Saba 2023). Visual analytics approaches combined with statistical methods improve detection accuracy by provid-

Table 1 Relevant studies on effective ways to filter bot responses from online surveys

Study	Sample size	Scope	Key detection methods	Key findings
Agans et al. (2024)	1,281	University Park, PA	Three-stage screening: eligibility questionnaire, targeted survey distribution, post-survey checks (inconsistencies, temporal clusters, attention checks)	15.4% valid responses; correlation patterns differed between bots and known humans
Caven et al. (2025)	2,115	Toronto, Canada	Data scoring approach with multiple filters; flagged responses failing 2+ criteria (inconsistent answers, 3-second completion clusters)	20.8% valid responses
Griffin et al. (2022)	1,251	Global	reCAPTCHA screening followed by post hoc filtering; tested raffle vs. guaranteed incentives	38.2% valid after filtering; 61.8% of reCAPTCHA-passed responses were bots
Hardesty et al. (2024)	1,624	Global	Flagged short completion times, duplicate accounts, inconsistent answers; ID verification before incentive payment	22.4% valid responses
Höhne et al. (2025)	4 bots	Germany	Experimental study programming bots with rule-based and LLM approaches; tested detection via item non-response and keystroke data	LLM bots maintained consistent personas; keystroke data detected advanced bots
Thompson and Utz (2025)	457	Salt Lake City, Utah	Combined hard flags (duplicate IPs, nonsensical text) and soft flags (location, duration, timing)	6% valid responses (94% identified as bots)
Xu et al. (2022)	3,200	Southern USA	Direct contact verification, suspicious timing analysis, response inconsistency checks	Contact verification most effective; timing patterns revealed bot clusters

ing researchers with more sophisticated identification tools (Chen et al. 2023).

Validation approaches include comparing response correlations against known human populations, but this requires appropriate reference data (Xu et al. 2022; Caven et al. 2025; Hardesty et al. 2024). Direct contact validation through phone or email verification provides strong confirmation but becomes impractical for large samples (Xu et al. 2022; Agans et al. 2024). Most critically, advanced language model-based bots can maintain consistent personas across responses, including demographic profiles and political views, making inconsistency-based detection increasingly unreliable (Höhne et al. 2025).

Despite these advances, significant gaps persist in providing practical, replicable frameworks. Most existing studies focus on either prevention strategies or post hoc detection in isolation, with limited guidance on establishing evidence-based filtering thresholds or systematically combining multiple approaches. Commercial solutions often operate as black boxes without transparency in their detection logic, while academic approaches struggle to provide generalizable methods that researchers can implement across different survey scenarios.

To address these gaps, we propose a novel framework to systematically assess the internal coherence of filtering decisions through machine learning classification and provide comprehensive robustness testing through threshold sensitivity and ablation analyses. Table 1 summarizes prior bot-detection studies, several of which combine more than one screening signal. However, existing approaches are often described at a high level, use context-specific heuristics without fully specified thresholds, and are frequently reported using non-comparable summary outcomes (for example, percent retained), which makes cross-study performance comparisons difficult without shared ground-truth labeling. In contrast, our framework is fully parameterized and designed as an end-to-end pipeline that can be deployed and tuned to fit a wide variety of use cases. We further demonstrate the statistical separability of the resulting rule-based decisions by training a model that reproduces filter decisions with a 0.94 F1 score and provides feature-level explanations. Together with threshold sensitivity and ablation analyses, this provides evidence-based guidance for deploying and tuning transparent bot filtering for online survey integrity.

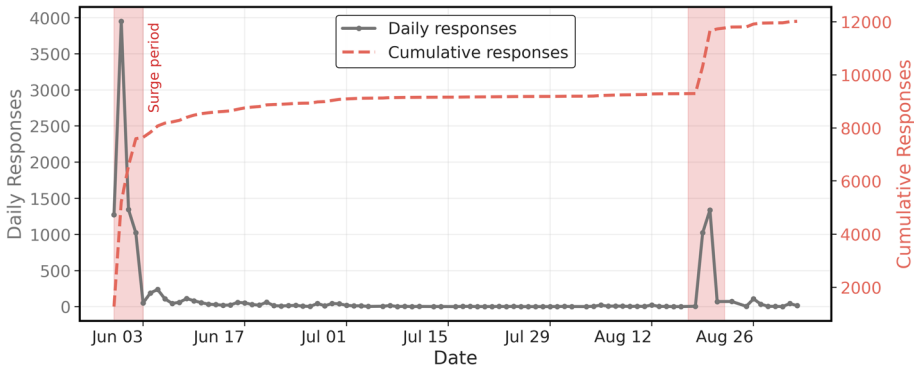

Fig. 1 Temporal distribution of submissions over the study period

Methods

Case Study

We designed an online survey to assess cyclists' attitudes toward bicycle infrastructure and wayfinding across demographic groups and usage patterns. We collected responses through the Qualtrics survey platform. The survey included 21 questions across 4 pages and was designed to take approximately 5–10 min to complete. Participants were to receive $7 compensation upon successful completion. We opened the survey on May 29, 2025, and closed it on August 31, 2025. Thus, we collected a total of 12,020 responses over the course of 94 days. Figure 1 shows the temporal distribution of responses.

Multi-filter Implementation

We applied a set of independent rules to identify and remove survey entries that demonstrate strong evidence of automation or low-quality completion. Each rule examines specific characteristics that online survey platforms can record during data collection, such as risk scores, interaction patterns, device information, and text responses. We defined explicit criteria for each filter and then tested whether these criteria are robust across different settings and combinations.

Figure 2 shows the methodology flowchart. Raw survey responses enter the pipeline, where each filter independently evaluates entries against its specific criteria. An entry is removed if it fails any single filter. We then validate the multi-filter pipeline through sensitivity analyses, ablation tests, a blind manual review, and a classification model.

We list all variables used in the multi-filter pipeline in Table 2. These fall into four categories: Response metadata, platform-generated risk scores, interaction patterns, and device metadata. The table also indicates which variables enter the rule-based filters, robustness checks, and model-based validation.

Risk scores are computed automatically when the survey is submitted. The reCAPTCHA score reflects how human-like session appears based on interaction patterns, while the fraud score is a composite risk indicator produced by the survey platform. Since these scores are computed outside our code, the exact scoring rules are not visible in the exported data, so we treat them as black-box inputs and document only their usage in the pipeline. Interaction data include counts of clicks across all pages, the time taken to advance through each page, and the total completion time from start to finish. These are logged by the survey platform without requiring special setup. Device metadata are obtained from the browser's user agent (UA) string and client-side scripts that capture screen resolution. The user agent also identifies the browser type and operating system, while the resolution provides the screen dimensions reported by the device. Finally, free-text fields, where respondents type open-ended answers, are examined for generic filler content or nonsensical patterns and a score is reported by the survey platform. Figure 3 presents the distributions of key variables used in bot detection.

We group the filters into four categories based on what they examine: platform-generated risk signals, identity and duplication, interaction patterns, and device consistency. Table 3 summarizes all filters along with their rationale.

Platform-Generated Risk Signals

Two risk scores helped us identify fraudulent submissions. The reCAPTCHA system assigns each session a score from 0.0 to 1.0 based on how human-like interaction patterns appear, with lower scores indicating a greater likelihood of bot activity. We adopted a threshold of 0.6, slightly more conservative than the Google-recommended default starting point of 0.5 (Google 2024), to minimize false negatives given the high volume of bot traffic. The distribution of our raw data supported this choice, showing a bimodal distribution where high-certainty human traffic stabilized above 0.6 (Fig. 3F).

Separately, the platform's fraud-detection module produces a composite score from 0 to 130 by examining IP reputation, device anomalies, and known problematic sources, where higher values signal a greater risk. We excluded entries with a fraud score above 30, a cutoff often cited as the

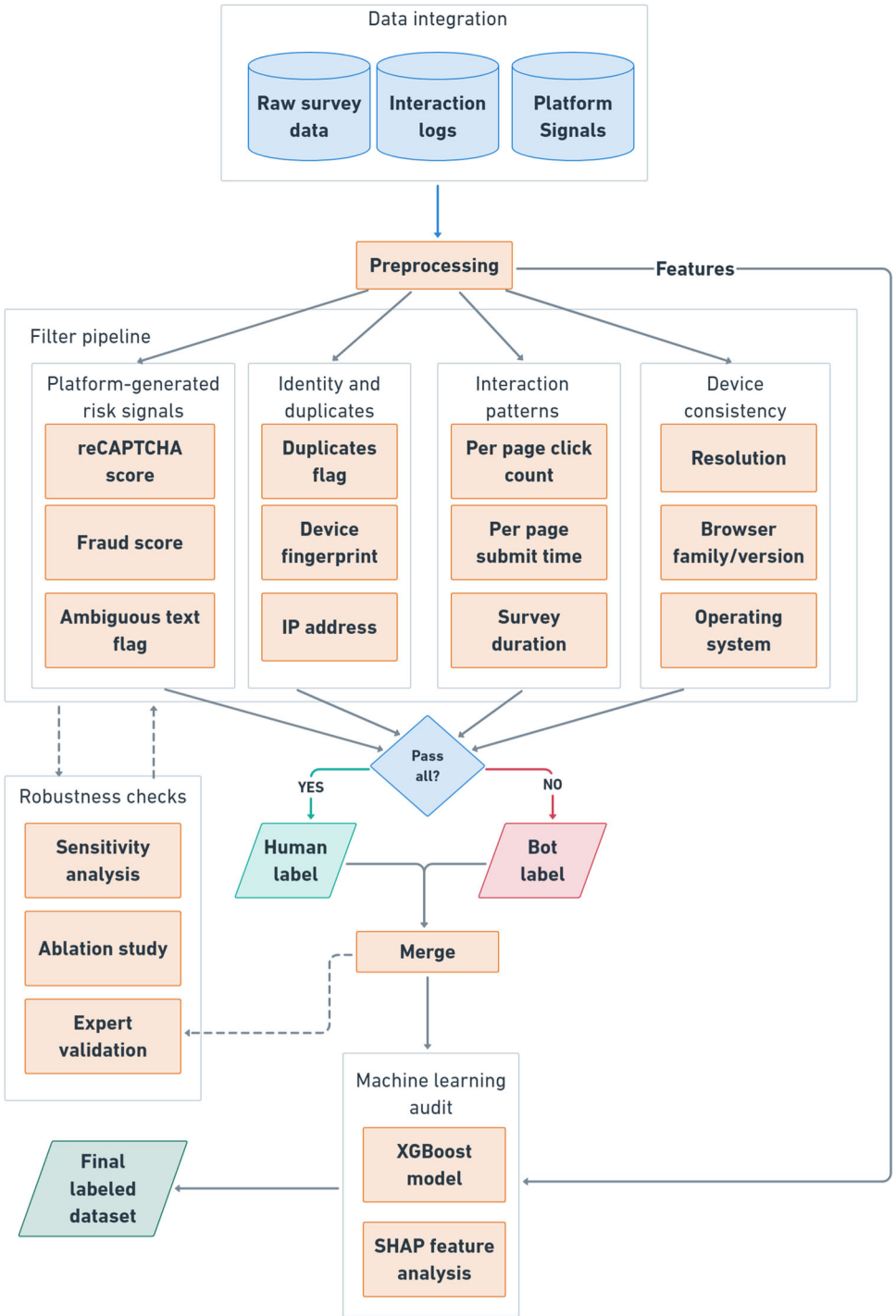

Fig. 2 Flowchart of the multi-filter bot-detection framework. Each filter operates independently on automatically recorded data or respondent-entered text. Entries failing any filter are removed. The final cleaned sample undergoes robustness checks to verify stability across threshold variations and filter combinations

lower bound for suspect activity in similar fraud-detection implementations (Bonett et al. 2024). This choice was further justified by the distribution of fraud risk scores in our sample, which showed a massive concentration of low-risk responses near 0, followed by distinct clusters of high-risk entries emerging at scores of 40, 75, and 110 (Fig. 3G).

We also used a variable generated by the survey platform that automatically evaluates open-ended text responses for incoherence and repetition, called the “Ambiguous Text" (Qualtrics 2025) flag. It identifies entries containing nonsensical or boilerplate content, such as repeated characters, filler terms, or copied text blocks. Since these patterns often arise from automated or inattentive submissions, we excluded all entries marked as True.

Table 2 Platform signals and metadata used in filtering, robustness checks, and model-based validation. A check mark indicates that the variable is used in the corresponding component of the method

Category	Variable	Description	Used in		
			Filter	Robustness	Model
Response metadata	Duration	Total time taken to finish the survey	✓	✓	✓
	Progress	Percent progress at submission			✓
	Finished	Indicator that the respondent finished the survey			✓
	Unanswered percentage	Share of items left unanswered (0–100)			✓
	Straightlining percentage	Proportion of items answered with identical options			✓
	Duplicate flag	Boolean indicator of duplicate submission	✓	✓	
	Duplicate score	Response similarity score (0–100)			✓
Platform-generated risk signals	reCAPTCHA score	Human-likeness score (0–1)	✓	✓	✓
	Fraud score	Composite platform fraud risk score (0–130)	✓	✓	✓
	Ambiguous-text flag	Indicator of incoherent free-text content	✓	✓	✓
Interaction patterns	Per-page click counts	Click counts for each survey page (1–4)			✓
	Total clicks	Count of clicks across all pages	✓	✓	
	Page-level submit time	Per-page completion times (Page 1–4)	✓	✓	✓
	Submit timing vector	Sequence of elapsed times between page advances	✓	✓	
Device metadata	IP address	Respondent IP address	✓	✓	
	Browser type	Browser family (e.g., Chrome, Safari)	✓	✓	✓
	Browser version	Browser version string	✓	✓	✓
	Operating system	OS family (e.g., Windows, Android, iOS)	✓	✓	✓
	Device type	Device class (mobile, tablet, desktop)	✓	✓	✓
	Screen resolution	Reported screen width × height in pixels	✓	✓	✓

Identity and Duplicates

We removed exact duplicate submissions from the sample that had been automatically flagged by the platform. We also examined IP addresses to detect responses from the same respondent. However, sharing an IP alone is not grounds for removal, as respondents in institutional settings, such as libraries or offices, may legitimately share network access. Instead, we removed entries only when the same IP appeared with identical device identifiers, which together indicated likely repeated attempts by the same source.

Interaction Patterns

Legitimate survey completion generates predictable interaction traces. A multi-page form requires clicking navigation buttons. Therefore, we removed any entries with zero recorded clicks throughout the session. The timing of page submissions also showed that automated tools often replay identical sequences across multiple entries, or they pause only on an initial challenge screen before advancing through the remaining pages nearly instantaneously. We removed entries showing either of these patterns.

Page-level submission behavior provides additional signals. We removed entries with extreme per-page submission times, defined as those falling in the top or bottom 5th percentiles of the page-completion distribution, with a lower threshold of 10 s to account for fraudulent responses. Similarly, total completion time separates likely automation (extremely fast) from abandoned sessions (extremely long), so we removed entries in the 5th and 95th percentiles of the total duration distribution. This symmetric 5% trimming follows common practice in survey and behavioral data cleaning, where completion time is used as a post hoc indicator of data quality, and the top and bottom 5% of the response-time distribution are excluded to retain a central 90% range of plausibly valid responses (Leiner 2019; Jiang et al. 2024).

Device Consistency

Finally, we removed entries where the screen resolution is unparsable, contains zero or undefined dimensions, or contradicts the claimed device type, such as desktop-scale dimensions reported from a phone or phone-scale dimensions reported from a desktop. We also removed submissions where the browser and operating system pairings were not feasible: Safari runs only on Apple systems, Internet Explorer runs only on Windows, and mobile browsers follow platform-

Table 3 Summary of bot-detection filters

Category	Filter name	Rationale
Platform-generated risk signals	*Low reCAPTCHA score*	Scores near 0.0 indicate bot-like interaction patterns
	High fraud score	High composite risk scores flag known bad actors, suspicious IPs, or device anomalies
	Response ambiguity	Generic fillers in free-text fields
Identity	*Duplicate identity*	Platform-flagged duplicates or same IP with matching device identifiers indicate repeated submissions
Interaction patterns	*Zero-clicks*	No clicks across entire survey is inconsistent with multi-page form completion
	Duplicate temporal pattern	Identical page-advance timing patterns or first-page-only timing suggest automated playback
	Extreme page duration	Unusually low or high per-page submit timing indicates instant or automated advancement
	Extreme survey duration	Very fast completions suggest automation; very slow completions suggest abandoned sessions
Device consistency	*Invalid resolution*	Unparsable, zero, or physically implausible screen dimensions for claimed device type
	Software incompatibility	Combinations that cannot co-exist (e.g., Safari on Android, Internet Explorer on macOS) reveal spoofed user agents
	User agent inconsistency	Individually plausible components that are inconsistent together (e.g., phone OS with desktop resolution)

specific engine rules. Violations indicate fabricated user agent strings that are common in bot traffic.

Robustness Checks

We tested the stability and internal consistency of the bot-detection multi-filter pipeline through component-level analysis and supervised learning validation. These checks examine the unique contribution of each filter and verify that the rule-based decisions form a coherent pattern distinguishable by machine learning algorithms.

Filter Contribution and Ablation

We conducted two diagnostic analyses to understand the impact of each filter within the pipeline. First, we compared the removal rate of each filter when applied in isolation to the raw data (independent application) versus its removal rate when applied in the fixed pipeline order (sequential application). This comparison reveals whether specific filters operate on unique subsets of the data or if they largely flag responses that would be flagged by earlier screening steps.

Second, we performed an ablation study to measure the marginal contribution of each filter to the final dataset. We removed one filter at a time and re-ran the entire pipeline with the remaining filters, recording the change in the final number of retained responses. Filters that result in a large increase in the retained sample when removed are identified as critical drivers of the detection process, while those that produce minimal change are identified as targeted safety nets that overlap with other criteria.

Manual Validation on a Blind Subsample

To provide an external check on the rule-based pipeline, we conducted a blind manual review on a stratified subsam-

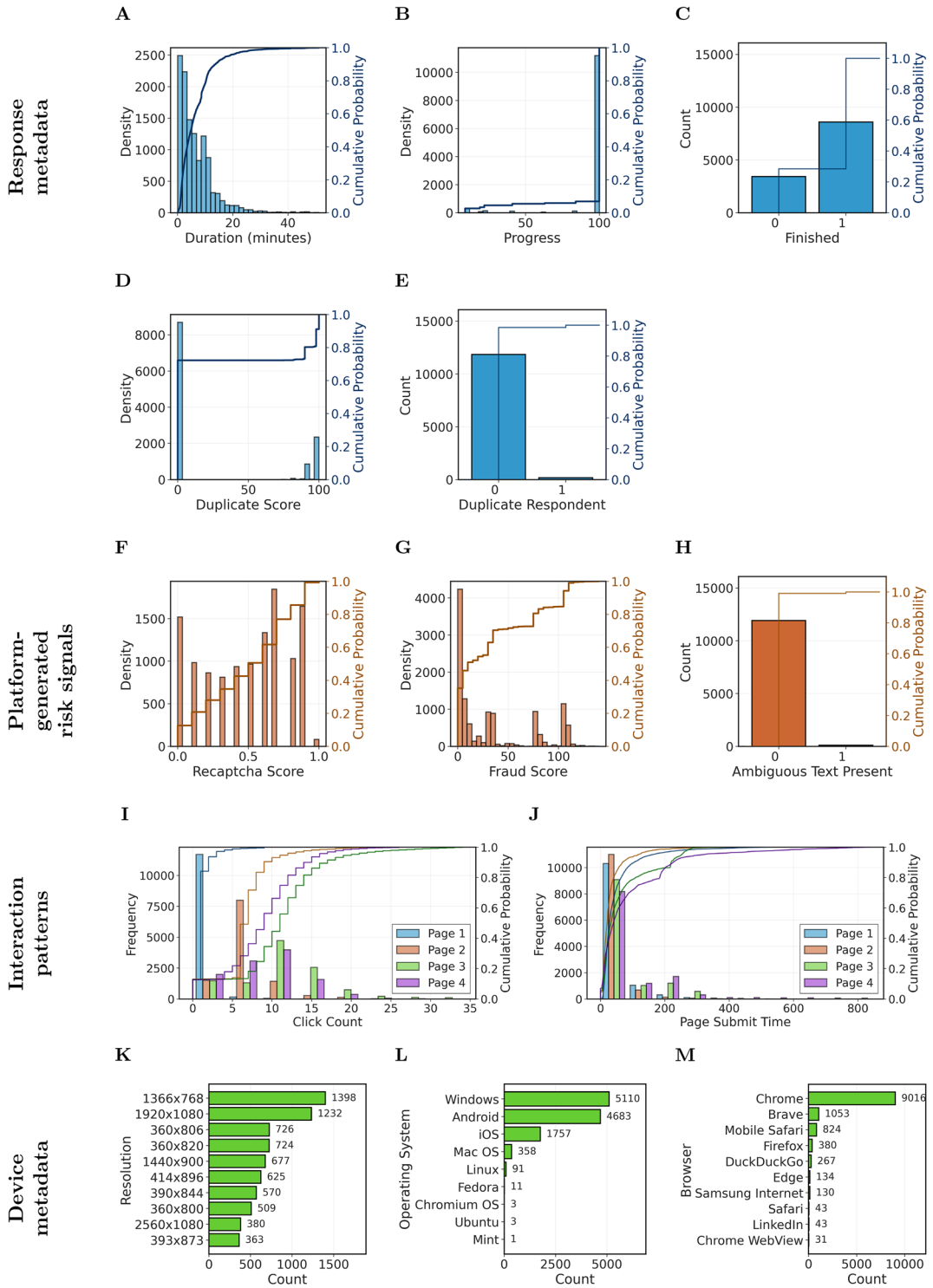

Fig. 3 Distributions of key filtering and inspection variables used in the multi-filter pipeline, grouped by category: *response metadata* **A** completion duration, **B** progress at submission, **C** finished indicator, **D** duplicate similarity scores, (E) duplicate respondent flag; *platform-generated risk signals* **F** reCAPTCHA scores, **G** fraud scores, **H** ambiguous text; *interaction patterns* **I** click counts on per page, **J** submit time per page; and *device metadata* **K** screen resolutions, **L** operating systems, and (M) browser families

ple of 100 responses: 50 that the pipeline retained and 50 that it removed. We created a blind file by removing the pipeline outcome and removing-filter fields, then an independent reviewer labeled each response as Human, Bot, or Unclear using page submit times (seconds), click counts, device metadata consistency, risk scores, and answer plausibility.

Model-Based Assessment

We converted the rule-based decisions into a supervised learning problem to assess the internal coherence of the filtering pipeline. We labeled entries that failed at least one independent filter as bot submissions (0), while entries that passed all filters and exhibited typical interaction patterns were labeled as human submissions (1). The resulting dataset comprised approximately 94.8% of cases labeled as bots and 5.2% labeled as humans by the pipeline.

We trained an extreme gradient boosting (XGBoost) classifier (Chen and Guestrin 2016) on these labeled cases using the same underlying features that the filters examine: risk scores, click counts, timing summaries, duration, device metadata, and text pattern flags, in addition to auxiliary variables such as completion status and unanswered percentages. Since the classifier is trained on rule-derived labels using features that largely overlap with the filter criteria, high classification agreement is expected. The value of this step, therefore, is in two outputs that the rule-based pipeline cannot produce, not in the accuracy itself. First, SHAP-based feature importance ranks all input features by their contribution to the classification decision given all other features. Unlike the ablation study, which can only rank the 11 existing filters by removal volume, SHAP can evaluate features that are not themselves used as standalone filters, identifying variables that carry discriminative power and may warrant inclusion in future pipeline implementations. Second, the classifier produces continuous predicted probabilities rather than binary pass-or-fail decisions, allowing researchers to rank responses by classification confidence and prioritize the most uncertain cases for manual review.

We performed a fivefold stratified cross-validation with a randomized hyperparameter search over 200 candidates (Table 4) using an 80/20 train-test split. As the joint search space is large, randomized search allows broad coverage under a fixed computational budget. Hyperparameters were tuned over the maximum tree depth, learning rate, number of estimators, row subsampling rate, and column sampling rate, as well as XGBoost regularization and tree-complexity controls (L1 and L2 regularization, split penalty, and minimum child weight), optimizing for the F1 score. The search identified the best-performing configuration as a maximum tree depth of 9, a learning rate of 0.05, 100 estimators, a row subsampling rate of 1.0, a column subsampling rate of 0.6, a minimum child weight of 5, a split penalty (γ) of 0.1, L1 regularization (α) of 0, and L2 regularization (λ) of 1. We evaluated the final model's performance using class-specific metrics (precision, recall, and F1 score).

To interpret the fitted XGBoost model, we computed Shapley Additive exPlanations (SHAP) using TreeSHAP for tree ensembles (Lundberg and Lee 2017). SHAP values were computed with a TreeExplainer in raw margin mode, such that SHAP contributions are expressed in log-odds units. These SHAP values explain the model output for the Human class (class 1): positive SHAP values increase the log-odds of Human, while negative values decrease the log-odds of Human. We used a background sample of $n = 2000$ observations to compute the expected value (base value) and evaluated SHAP on a stratified sample of $n = 1000$ held-out test observations. We also examined non-linear effects and context dependence via SHAP dependence plots for the highest-ranked features.

Results

Multi-filter Pipeline Effectively Isolates Valid Responses by Targeting Distinct Bot Indicators

The initial screening, utilizing platform-generated signals, identified substantial bot activity. As detailed in Table 5, the *low reCAPTCHA score* filter removed 50.60% of entries falling below the 0.6 threshold. This cutoff effectively eliminated the bottom half of the raw distribution, corresponding to the 50.6 percentile (Fig. 4A). Subsequently, the *high fraud score* filter removed 44.76% of the remaining sample by setting the threshold at 30, removing the top tail of the risk distribution (Fig. 4B). Lastly, the *response ambiguity* filter caught 1.28% of the remaining entries that contained generic filler patterns (Fig. 4C).

Following the risk score assessment, the identity and interaction filters targeted sophisticated bots that bypassed the initial screening. The *duplicate identity* filter eliminated 7.15% of the surviving sample (Fig. 4D). The interaction pattern filters then removed a significant portion of the remaining data: 23.88% of entries recorded zero interaction events (Fig. 4E), while 2.54% exhibited identical page-advance timing vectors (Fig. 4F). Most notably, the *extreme page duration* filter removed 59.83% of the remaining entries. This indicates that a large volume of submissions passed the reCAPTCHA check but failed to generate human-like reading and response speeds (Fig. 4G). The *extreme survey duration* rule then further removed 10.16% of entries with implausibly fast or slow total completion times (Fig. 4H).

In the final device consistency filtering stage, the *invalid resolution* filter removed 0.47% (Fig. 4I), while the *software incompatibility* filter removed 1.89% of the remaining

Table 4 XGBoost hyperparameter search space and best-performing configuration. To manage computation, we used standard parameter ranges and used a randomized search strategy

Hyperparameter	Search space	Best value
Maximum tree depth	{3, 5, 7, 9}	9
Learning rate	{0.01, 0.05, 0.1}	0.05
Number of estimators	{100, 200, 300, 800, 1500, 2500}	100
Row subsampling (subsample)	{0.6, 0.8, 1.0}	1.0
Column subsampling (colsample_bytree)	{0.6, 0.8, 1.0}	0.6
Minimum child weight (min_child_weight)	{1, 5, 10}	5
Split penalty (γ)	{0, 0.1, 0.5, 1}	0.1
L1 regularization (α)	{0, 0.001, 0.01, 0.1, 1}	0
L2 regularization (λ)	{0.1, 1, 5, 10, 50}	1

Table 5 Summary of filtering thresholds and exclusion statistics

Category	Filter	Threshold		Percentile		Filtered	Cumulative%
		LB	UB	LB	UB		
Platform-generated risk signals	*Low reCAPTCHA score*		**0.6**		50.6	6,082	50.60
	High fraud score	**30**		55.2		2,658	72.71
	Response ambiguity (0/1)		**0**		98.7	42	73.06
Identity	*Duplicate identity*		**0**		73.5	859	80.21
Interaction patterns	*Zero-clicks*	**1**		23.9		568	84.94
	Duplicate temporal pattern (0/1)		**0**		97.5	46	85.32
	Extreme page duration (s)	9	27	**5**	**95**	1,056	94.11
	Extreme survey duration (s)	147	1051	**5**	**95**	72	94.71
Device consistency	*Invalid resolution* (0/1)		**0**		99.5	3	94.73
	Software incompatibility (0/1)		**0**		98.1	12	94.83
	User agent inconsistency (0/1)		**0**		99.7	2	94.84

LB and *UB* denote the lower bound and upper bound of the applied criteria, respectively. Thresholds for the *extreme page duration* and *extreme survey duration* filters were derived from the 5th and 95th statistical percentiles of the distribution. All other filters employed fixed value cutoffs or binary flags. The "Percentile" shows where each threshold falls in the distribution at that filtering stage. Bold values indicate the parameter (either a fixed threshold or a statistical percentile) used to establish the exclusion rule for each filter. The final column displays the cumulative percentage of the total sample removed from the original 12,020 responses

entries (Fig. 4J). Additionally, *user agent inconsistency*, which detects clear anomalies in the user agent strings, removed 0.32% (Fig. 4K).

Applying all filters sequentially reduced the sample from 12,020 raw responses to 620 valid entries, resulting in a total reduction of 94.8% (Table 5). We provide the specific parameterization that drives these removals in Table 5.

Expert Review Quantifies Agreement Between Automated and Human Classification

To provide an independent reference point for the multi-filter pipeline, we conducted a blind manual review of a stratified subsample of 100 responses (50 retained, 50 removed). Since no external ground truth exists for this dataset, neither the pipeline nor the expert can be assumed definitively correct. This comparison, therefore, quantifies the level of agreement between the two independent classification approaches and characterizes the types of responses on which they disagree.

The manual validation demonstrated strong alignment between the rule-based pipeline and human judgment (Fig. 5). Among the 90 responses with a definitive expert label, the pipeline agreed with the reviewer in 88.9% of cases (80 out of 90). The ten disagreements consisted of three responses that the pipeline retained but the expert labeled as bots, and seven responses that the pipeline removed but the expert labeled as human. An additional ten cases were considered too ambiguous for the expert to definitively label.

Breaking down the disagreements by filter stage reveals which filters are associated with the most frequent discrepancies between automated and expert classification (Table 6). The responses that the pipeline removed but the expert labeled as human were primarily concentrated in the low reCAPTCHA stage (4 cases) and the extreme page-duration stage (2 cases). Conversely, the high fraud-score filter produced zero disagreements of this type. As reCAPTCHA and timing filters drive the absolute majority of data exclusion

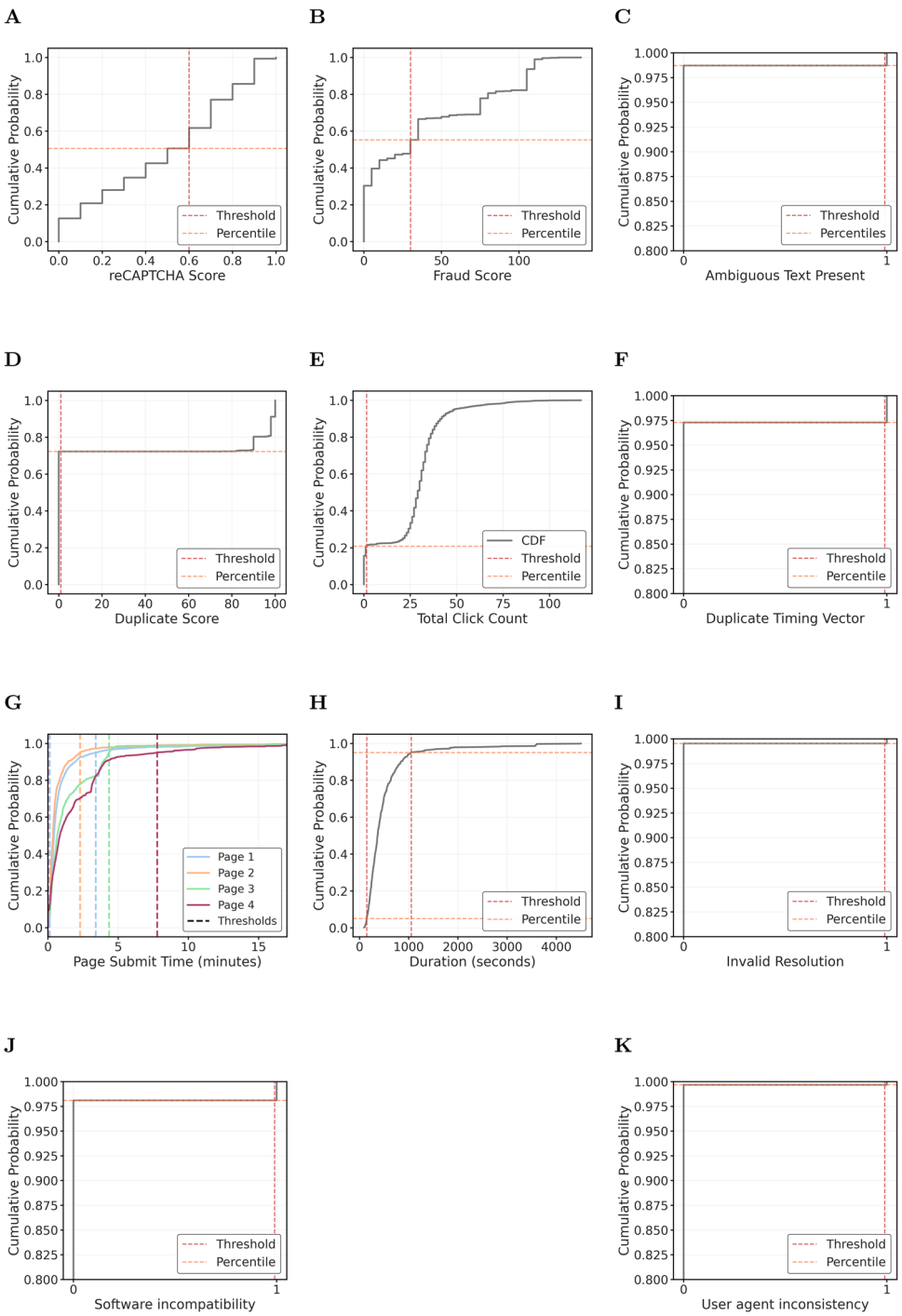

Fig. 4 Cumulative distribution functions (CDFs) of filtering variables applied in the pipeline. Dashed lines indicate the exclusion thresholds: vertical lines denote specific value cutoffs (with binary indicators encoded as 0 for valid/false and 1 for invalid/true), and horizontal lines denote the corresponding exclusion percentiles. The subfigures display: **A** reCAPTCHA score, **B** fraud score, **C** ambiguous text indicator, **D** duplicate score, **E** total click count, **F** duplicate temporal pattern indicator, **G** per-page submit timing (showing lower and upper bounds), **H** total survey duration, **I** invalid resolution indicator, **J** software incompatibility indicator, and **K** user agent inconsistency indicator

(Table 5), understanding the sources of disagreement at these stages is important for calibrating thresholds in practice.

Platform-Generated Signals are Insufficient to Filter Out Bot Responses

We analyze the difference between the impact of applying the filter on the removal rate measured against the original data and the remaining data in Fig. 6. While platform-generated signals like reCAPTCHA account for the bulk of raw removals, the interaction and device pattern filters still provide necessary screening. For example, the *extreme page duration* filter accounts for 8.79% of the original raw sample, yet it is responsible for removing 59.83% of the submissions that successfully passed the initial platform screening (Fig. 6). Similarly, the *duplicate identity* and *zero-clicks* filters show low removal rates from the original data but eliminate 26.53% and 23.88% of the remaining traffic, respectively.

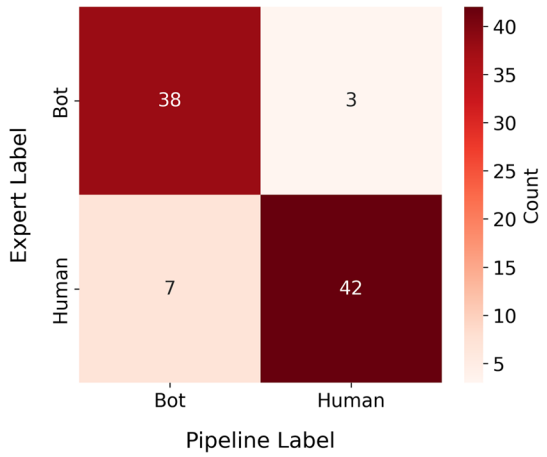

Fig. 5 Confusion matrix comparing the multi-filter pipeline's decisions against the independent expert labels for the 90 responses with a definitive classification

This indicates that a substantial volume of sophisticated bot traffic is only detectable through these secondary behavioral checks.

We also compare the number of responses each filter removes in the actual pipeline order (sequential) versus the number it would remove if applied alone to the raw data (independent) in Fig. 7A. Most filters show substantially larger independent effects than sequential effects, which shows that many automated submissions trigger multiple flags across different categories.

To identify the most critical components of the multi-filter pipeline, we conducted an ablation test in Fig. 7B. Turning off the *low reCAPTCHA score* filter increases the retained sample from 620 to 1734 respondents (+9.3 percentage points [pp] of the original pipeline). Disabling the *extreme page duration* filter yields a similarly large change (1528 retained, +7.6 pp). In contrast, removing the *high fraud score*, *zero-clicks*, or *duplicate identity* filters produces more moderate gains (1.5—3.0 pp). The *response ambiguity*, *software incompatibility*, and *user agent inconsistency* filters change the retained proportion by less than 0.1 pp when ablated, indicating that they could function as targeted safety nets for specific edge cases rather than primary drivers of the removal rate.

These findings expose a gap in standard platform defenses. Relying solely on reCAPTCHA, fraud scores, text analysis, and duplicates would have yielded a sample of 2379 responses. However, our interaction and device filters flagged 1759 of these as invalid. Therefore, 73% of the data approved by the platform's default tools consisted of fraudulent or invalid submissions that were only identified by examining interaction and device patterns.

Supervised Learning Recovers Filter Decisions and Infers Key Variable Relationships

The classifier distinguished between filtered and retained responses with very high F1 score on a held-out test set (Table 7). Given the class imbalance, we prioritize the human-class performance: an F1 score of 0.94, driven by a recall of 0.98 and a precision of 0.90. Only 2 of 124 retained humans are misclassified as bots, while 13 of 2280 removed responses are classified as human (Fig. 8A). In addition to categorical predictions, the classifier assigns a continuous predicted probability of being human to each response. Unlike the rule-based pipeline, which treats any single-filter failure as equivalent to failing all filters, the predicted probabilities allow researchers to rank responses by classification confidence and focus limited manual review resources on the most uncertain cases. Bot detection is also strong, with a precision of 0.9991 and a recall of 0.9943. The ROC-AUC on the test set is 0.9978.

The SHAP values reveal which features most strongly influence the model's predictions (Fig. 8B). The reCAPTCHA score is the most influential variable. Page-level timing measures also play a significant role, specifically the time of the last click on page 2, which ranked second and helps distinguish between immediate advancement and human deliberation. These two were followed by the fraud score, duplicate respondent score, and the overall survey duration.

Table 6 Detailed breakdown of the 100-response blind validation sample by the specific pipeline decision

Pipeline decision	*n*	Expert Labels		
		Human	Bot	Unclear
Retained (passed all filters)	50	42	3	5
Removed by low reCAPTCHA score	27	4	20	3
Removed by high fraud score	11	0	10	1
Removed by duplicate identity	1	0	1	0
Removed by zero-clicks	5	1	4	0
Removed by extreme page duration	5	2	2	1
Removed by extreme survey duration	1	0	1	0
Total	100	49	41	10

Rows indicate the pipeline's outcome, and columns report the independent expert's label

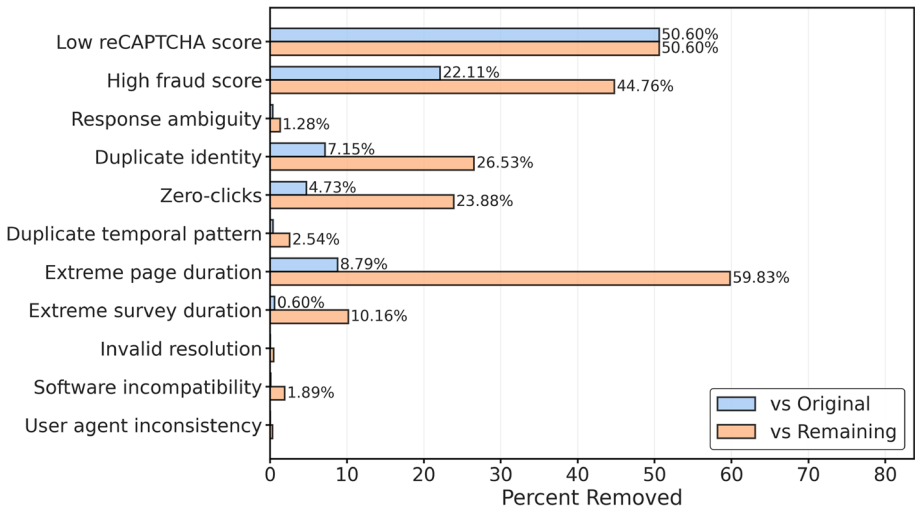

Fig. 6 Removal rates for each filter calculated relative to the original and remaining sample at that stage

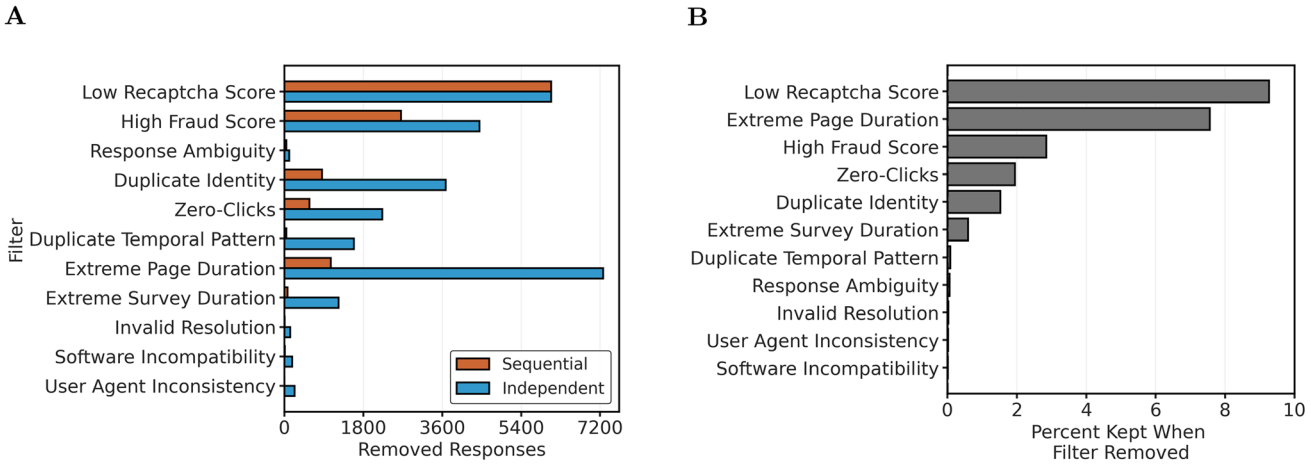

Fig. 7 Per-filter sensitivity: **A** Responses removed by each filter in the actual pipeline order versus if applied alone to the raw data, and **B** retained sample size when each filter is disabled in turn

Table 7 XGBoost classification performance on held-out test set

Class	Precision	Recall	F1 score	Accuracy	ROC-AUC	Support
Bot (0)	0.9991	0.9943	0.9967	–	–	2280
Human (1)	0.9037	0.9839	0.9421	–	–	124
Overall				0.9938	0.9978	2404

The mean absolute SHAP importance for each feature is shown in parentheses next to the feature name in Fig. 8B. In terms of direction, higher reCAPTCHA scores push predictions toward the human class, while higher fraud and duplicate scores push strongly toward the bot class. Timing features and duration show mixed patterns that indicate both classes depend on whether interactions are unusually fast or slow. The finished status shows that completed responses increase the probability of being classified as human. This feature importance ranking identifies variables that the ablation study cannot evaluate because they are not standalone filters. Page 2 last click time, which records the timing of the last interaction event on a page and is distinct from the page submit times used in the extreme page duration filter, ranks second overall, above the fraud score. Similarly, the finished status indicator and Page 4 last click time appear in the top ten despite not being used as filter criteria. These rankings suggest that per-page click timing and completion status carry substantial discriminative power and are candidate variables for future filtering pipeline implementations. The ablation study (Fig. 7B), however, measures a different quantity, namely the marginal effect on sample size when each existing filter is removed, and therefore cannot provide this type of guidance for identifying new filtering signals.

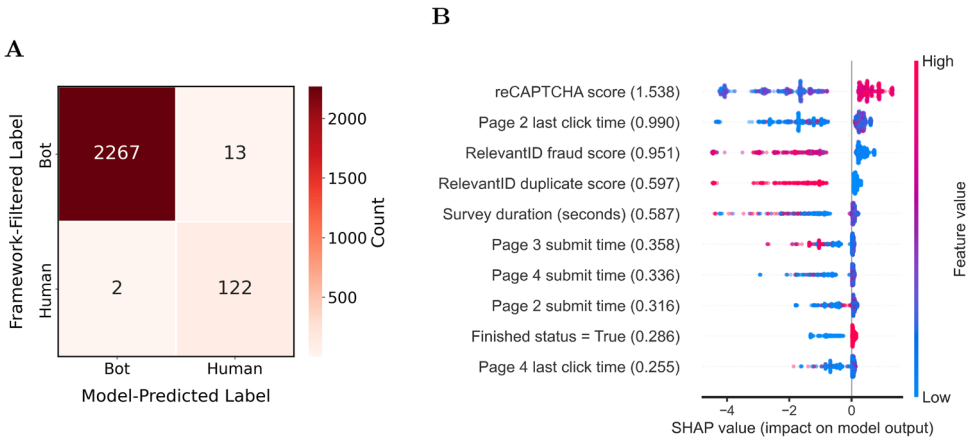

Fig. 8 XGBoost validation results: **A** Confusion matrix, and (B) SHAP beeswarm plot for the top ten features. In **B**, each dot represents one observation; color encodes the feature value (red = high, blue = low), and the horizontal position shows the SHAP contribution to the model output (positive values increase the likelihood of Human and negative values increase the likelihood of Bot). Values shown in parentheses next to each feature name denote the corresponding mean absolute SHAP importance, mean(|SHAP|), computed over the evaluation sample

Feature Dependence Reveals Risk Score Thresholds and Non-monotonic Timing Patterns

Global feature rankings often visually mix high and low values on both sides of zero, which typically indicates a non-monotonic effect (Fig. 8B). To directly examine these patterns, we provide SHAP dependence plots for the top-ranked features (Fig. 9).

The dependency plots show how the sign and magnitude of each feature's contribution vary across its observed range. For the platform risk signals, the patterns are largely directional but non-linear (step functions). In particular, reCAPTCHA exhibits a strong separation: for scores below roughly 0.6, the contributions are consistently negative. Similarly, the fraud score shows a drop in contribution at high values: scores over 30 are strongly negative.

In contrast, the behavioral timing features exhibit non-monotonic effects. For Page 2 last click time, the largest negative contributions occur at very short values (rushing). However, very long times also frequently correspond to negative SHAP values, while the positive contributions are concentrated in a middle range. Survey duration displays a similar structure: shorter durations are negative on average, a central plausible range becomes positive, and extreme long durations again contain negative contributions. This confirms that the model is learning to penalize both too fast (bot-like) and too slow (abandoned/idler) behaviors.

Discussion

This study demonstrates that comprehensive bot detection in incentivized online surveys requires a layered approach that extends beyond platform-provided signals. Our novel multi-filter pipeline successfully identified patterns of fraudulent activity that evaded standard detection mechanisms, with two key findings emerging from our analysis. First, platform-generated risk scores effectively screen the majority of unsophisticated bot traffic, with reCAPTCHA and fraud scores together removing over 70% of submissions. Yet, a significant portion of possible bot responses is only removed by our novel metrics. Thus, interaction pattern analysis proves essential for detecting sophisticated automation that mimics human-like risk profiles but fails to replicate natural survey completion behaviors. Second, device consistency checks serve as targeted safety nets that catch edge cases involving fabricated user agent strings or implausible hardware configurations.

Transferability

A major advantage of this multi-filter pipeline is that it can be readily applied in a wide range of data collection settings. Although our case study uses Qualtrics, the filter logic is modular and does not strictly rely on proprietary fields. To deploy this pipeline elsewhere, researchers can map the criteria in Table 3 to their platform's available metadata. Most survey tools provide or log four core signal types: (1) risk scores or challenge outputs (e.g., reCAPTCHA), (2) duplicate markers (e.g., cookies or repeated IP/device combinations), (3) interaction traces (click counts and page timings), and (4) device metadata (browser, OS, device type, and screen size). If a composite fraud score is unavailable, the remaining filters still apply, and researchers can calibrate their specific thresholds using the sensitivity and ablation procedures detailed in "Robustness Checks".

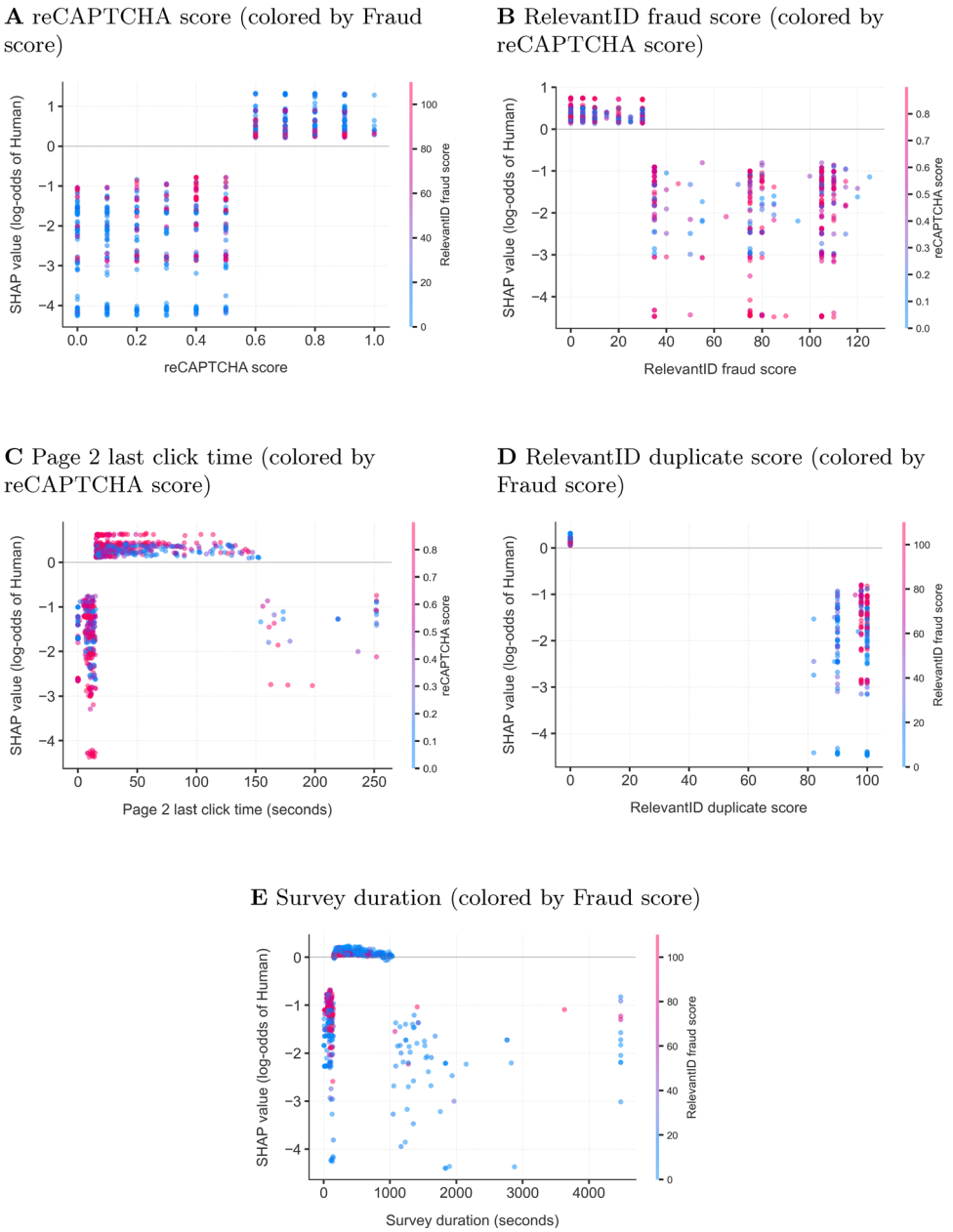

Fig. 9 SHAP dependence plots for the top five features. SHAP values are computed in raw margin mode and therefore are expressed in log-odds of the Human class (class 1). Points are colored by a selected companion variable to visualize context dependence and potential non-linear effects

Study Limitations and Future Work

While our framework provides a transparent, replicable foundation for bot detection that can be readily adapted to diverse online survey contexts, certain limitations remain. First, some platform-generated risk signals are vendor-produced scores, so their internal scoring logic cannot be fully audited from exported survey data. Second, our analysis focuses on a single survey platform (Qualtrics) in one research domain (transportation). While the underlying principles should generalize, threshold values may require calibration to fit varying survey lengths, incentive structures, or participant populations. Future work should evaluate how well these parameters transfer to other fields.

Third, the rapidly evolving nature of bot technology means that detection strategies require continuous updating. Our device consistency checks assume that certain browser-operating system combinations are technically infeasible, but future browser engines or virtualization technologies may enable previously impossible configurations. Similarly, advances in large language models may enable bots to generate more contextually appropriate open-ended responses

that evade ambiguous text detection. Researchers should regularly reassess their filtering criteria against emerging bot capabilities and consider incorporating additional behavioral signals where available.

Finally, our symmetric percentile trimming approach (5th and 95th percentiles) for duration filtering may exclude some legitimate extreme responses. Future work could explore asymmetric thresholds or more sophisticated outlier detection methods that account for survey-specific characteristics.

Conclusion

This study presents a transparent and generalizable multi-filter pipeline for securing online survey data. In contrast to commercial solutions with proprietary detection logic, our pipeline enables researchers to explicitly define and adjust the criteria for valid responses. While standard platform signals like reCAPTCHA serve as a useful first line of defense, our results demonstrate that supplementary behavioral and device metrics are critical. In our case study, 73% of the responses that passed standard platform filters were subsequently identified as invalid through interaction and device pattern analysis. Given that these filters operate on widely available survey metadata, the multi-filter pipeline is readily transferable to other research contexts without requiring specialized commercial tools.

To establish ground-truth performance and validate our multi-filter pipeline, we conducted a blind manual review by an independent expert on a stratified subsample. The rule-based pipeline demonstrated strong alignment with human judgment, achieving an 88.9% agreement rate. Since no external ground truth exists, this figure represents concordance between two independent classification approaches rather than a measure of absolute accuracy. Nevertheless, the high level of agreement, combined with the observation that most disagreements are concentrated at specific filter stages, provides evidence that the pipeline produces reasonable and interpretable decisions.

Furthermore, we trained a supervised machine learning classifier to audit the internal coherence of the multi-filter pipeline. The model independently reproduced the filtering decisions, achieving an F1 score of 0.94 for the human class and confirming that the filtered groups are statistically separable based on observed behaviors. SHAP analysis demonstrated that the model's most influential features align closely with our filtering criteria. This proves that the rules capture genuine behavioral differences rather than relying on arbitrary thresholds, representing a novel application of explainable machine learning to audit survey bot detection.

Our work offers three distinct contributions. First, we provide a complete, open-source methodology with tunable parameters that researchers can adapt across different survey platforms and domains. Second, we establish robust, evidence-based criteria through threshold sensitivity analysis, filter ablation, and expert validation. Third, we introduce supervised learning as a novel audit mechanism that independently verifies filtering decisions using interpretable feature importance. This framework provides a practical template for improving data quality, bridging the gap between isolated detection techniques and actionable implementation.

Author Contributions Conceptualization: Jimi Oke, Eleni Christofa, Mohammed Abdalazeem. Methodology: Jimi Oke, Mohammed Abdalazeem, Eleni Christofa, Andrew Ruger. Formal analysis and investigation: Mohammed Abdalazeem, Andrew Ruger, Jimi Oke, Eleni Christofa. Software: Mohammed Abdalazeem, Andrew Ruger. Writing—original draft preparation: Andrew Ruger, Mohammed Abdalazeem. Writing—review and editing: Mohammed Abdalazeem, Jimi Oke, Eleni Christofa. Funding acquisition: Eleni Christofa, Jimi Oke. Resources: Jimi Oke. Supervision: Jimi Oke, Eleni Christofa.

Funding This research was funded by the US Department of Transportation via the New England University Transportation Center (Grant Number: 69A35523483) and the Armstrong Fund for Science.

Data Availability The code underlying the multi-filter bot detection framework is available in a dedicated GitHub repository at https://github.com/narslab/survey-bot-detection. The survey datasets analyzed during the current study are not publicly available due to privacy restrictions regarding human participants.

Declarations

Conflict of interest The authors declare no conflict of interest.

Ethical approval This study was approved by the Institutional Review Board at the University of Massachusetts Amherst (Protocol #5917, approved May 15, 2025). All survey participants provided informed consent prior to participation.

References

Agans JP, Schade SA, Hanna SR, Chiang S-C, Shirzad K, Bai S (2024) The inaccuracy of data from online surveys: a cautionary analysis. Qual Quant 58(3):2065–2086. https://doi.org/10.1007/s11135-023-01733-5

Ahn L, Cathcart W (2009) Teaching computers to read: google acquires reCAPTCHA. Official Google Blog. https://googleblog.blogspot.com/2009/09/teaching-computers-to-read-google.html. Accessed 19 Nov 2025

Bonett S, Lin W, Sexton Topper P, Wolfe J, Golinkoff J, Deshpande A, Villarruel A, Bauermeister J (2024) Assessing and improving data integrity in web-based surveys: comparison of fraud detection systems in a COVID-19 study. JMIR Form Res 8:47091. https://doi.org/10.2196/47091. arXiv:3821.4962

Buchanan EM, Scofield JE (2018) Methods to detect low quality data and its implication for psychological research. Behav Res Methods 50(6):2586–2596. https://doi.org/10.3758/s13428-018-1035-6. arXiv:2954.2063

Caven I, Yang Z, Okrainec K (2025) It's raining bots: how easier access to internet surveys has created the perfect storm. BMJ Open Qual. https://doi.org/10.1136/bmjoq-2024-003208

Chen T, Guestrin C (2016) XGBoost: a scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. ACM, pp 785–794. https://doi.org/10.1145/2939672.2939785. arXiv:1603.02754. Accessed 14 Jul 2025

Chen AT, Komi M, Bessler S, Mikles SP, Zhang Y (2023) Integrating statistical and visual analytic methods for bot identification of health-related survey data. J Biomed Inform 144:104439. https://doi.org/10.1016/j.jbi.2023.104439

Conrad FG, Couper MP, Tourangeau R, Zhang C (2017) Reducing speeding in web surveys by providing immediate feedback. Surv Res Methods 11(1):45–61. https://doi.org/10.18148/srm/2017.v11i1.6304. arXiv:3174.5400

Dennis SA, Goodson BM, Pearson CA (2020) Online worker fraud and evolving threats to the integrity of MTurk data: a discussion of virtual private servers and the limitations of ip-based screening procedures. Behav Res Account 32(1):119–134. https://doi.org/10.2308/bria-18-044

Dupuis M, Meier E, Cuneo F (2019) Detecting computer-generated random responding in questionnaire-based data: a comparison of seven indices. Behav Res Methods 51(5):2228–2237. https://doi.org/10.3758/s13428-018-1103-y

Fang K (2020) Surveying silicon valley on cycling, travel behavior, and travel attitudes. Technical report, Mineta Transportation Institute. https://doi.org/10.31979/mti.2020.1947. https://scholarworks.sjsu.edu/mti_publications/315/. Accessed 15 Nov 2025

Fielding N, Lee R, Blank G (2017) Online research methods in the social sciences: an editorial introduction. In: Handbook of online research methods, 2nd ed. Sage Publications, pp 3–16. https://www.researchgate.net/publication/321884086_Online_research_methods_in_the_social_sciences_An_editorial_introduction. Accessed 2025-07-07

Forsyth A (2010) Measuring walking and cycling using the PABS (pedestrian and bicycling survey) approach: a low-cost survey method for local communities. Mineta Transportation Institute

Godinho A, Schell C, Cunningham JA (2020) Out damn bot, out: recruiting real people into substance use studies on the internet. Subst Abuse 41(1):3–5. https://doi.org/10.1080/08897077.2019.1691131. arXiv:3182.1108

Google (2024) reCAPTCHA v3. Google for developers. https://developers.google.com/recaptcha/docs/v3. Accessed 19 Nov 2025

Griffin M, Martino RJ, LoSchiavo C, Comer-Carruthers C, Krause KD, Stults CB, Halkitis PN (2022) Ensuring survey research data integrity in the era of internet bots. Qual Quant 56(4):2841–2852. https://doi.org/10.1007/s11135-021-01252-1

Hardesty JJ, Crespi E, Sinamo JK, Nian Q, Breland A, Eissenberg T, Kennedy RD, Cohen JE (2024) From doubt to confidence-overcoming fraudulent submissions by bots and other takers of a web-based survey. J Med Internet Res 26:60184. https://doi.org/10.2196/60184. arXiv:3968.0887

Hardinghaus M, Papantoniou P (2020) Evaluating cyclists' route preferences with respect to infrastructure. Sustainability 12(8):3375. https://doi.org/10.3390/su12083375

Höhne JK, Claassen J, Shahania S, Broneske D (2025) Bots in web survey interviews: a showcase. Int J Mark Res 67(1):3–12. https://doi.org/10.1177/14707853241297009

Irish K, Saba J (2023) Bots are the new fraud: A post-hoc exploration of statistical methods to identify bot-generated responses in a corrupt data set. Personal Individ Differ 213:112289. https://doi.org/10.1016/j.paid.2023.112289

Jiang M, Yasui K, Yugami K (2024) Working from home, job tasks, and productivity. Telecommun Policy 48(8):102806. https://doi.org/10.1016/j.telpol.2024.102806

Kennedy R, Clifford S, Burleigh T, Waggoner PD, Jewell R, Winter NJG (2020) The shape of and solutions to the MTurk quality crisis. Polit Sci Res Methods 8(4):614–629. https://doi.org/10.1017/psrm.2020.6

Leiner DJ (2019) Too fast, too straight, too weird: non-reactive indicators for meaningless data in internet surveys. Surv Res Methods 13(3):229–248. https://doi.org/10.18148/srm/2019.v13i3.7403

Lundberg S, Lee S-I (2017) A unified approach to interpreting model predictions. https://doi.org/10.48550/arXiv.1705.07874. arXiv:1705.07874

Petrik O, Silva JDAE, Moura F (2016) Stated preference surveys in transport demand modeling: disengagement of respondents. Transp Lett 8(1):13–25. https://doi.org/10.1179/1942787515Y.0000000003

Plesner A, Vontobel T, Wattenhofer R (2024) Breaking reCAPTCHAv2. In: 2024 IEEE 48th annual computers, software, and applications conference (COMPSAC). pp 1047–1056. https://doi.org/10.1109/COMPSAC61105.2024.00142. arXiv:2409.08831. Accessed 03 Jul 2025

Ponto J (2015) Understanding and evaluating survey research. J Adv Pract Oncol 6(2):168–171 arXiv:2664.9250

Pozzar R, Hammer MJ, Underhill-Blazey M, Wright AA, Tulsky JA, Hong F, Gundersen DA, Berry DL (2020) Threats of bots and other bad actors to data quality following research participant recruitment through social media: cross-sectional questionnaire. J Med Internet Res 22(10):23021. https://doi.org/10.2196/23021

Qualtrics (2025) Response quality. qualtrics support. https://www.qualtrics.com/support/survey-platform/survey-module/survey-checker/response-quality/. Accessed 14 Nov 2025

Regmi PR, Waithaka E, Paudyal A, Simkhada P, Teijlingen E (2016) Guide to the design and application of online questionnaire surveys. Nepal J Epidemiol 6(4):640–644. https://doi.org/10.3126/nje.v6i4.17258

Searles A, Nakatsuka Y, Ozturk E, Paverd A, Tsudik G, Enkoji A (2023) An empirical study and evaluation of modern CAPTCHAs. In: Proceedings of the 32nd USENIX Security Symposium, USENIX Association, pp 3081–3097. https://www.usenix.org/conference/usenixsecurity23/presentation/searles

Singh S, Sagar R (2021) A critical look at online survey or questionnaire-based research studies during. Asian J Psychiatry COVID-19 65:102850. https://doi.org/10.1016/j.ajp.2021.102850

Sivakorn S, Polakis I, Keromytis AD (2016) I am robot: (deep) learning to break semantic image CAPTCHAs. In: 2016 IEEE European symposium on security and privacy (EuroS&P), pp. 388–403. https://doi.org/10.1109/EuroSP.2016.37. https://ieeexplore.ieee.org/document/7467367. Accessed 19 Nov 2025

Storozuk A, Ashley M, Delage V, Maloney EA (2020) Got bots? Practical recommendations to protect online survey data from bot attacks. Quant Methods Psychol 16(5):472–481. https://doi.org/10.20982/tqmp.16.5.p472

Teitcher JEF, Bockting WO, Bauermeister JA, Hoefer CJ, Miner MH, Klitzman RL (2015) Detecting, preventing, and responding to "fraudsters" in internet research: ethics and tradeoffs. J Law Med Ethics 43(1):116–133. https://doi.org/10.1111/jlme.12200

Thompson AD, Utz RL (2025) Online surveys: lessons learned in detecting and protecting against insincerity and bots. Qual Quant 59(1):23–39. https://doi.org/10.1007/s11135-024-01973-z

Vasantha Raju N (2016) Online survey tools, a case study of Google forms. In: National conference on scientific, computational & information research trends in engineering, GSSS-IETW, Mysore

Xu Y, Pace S, Kim J, Iachini A, King LB, Harrison T, DeHart D, Levkoff SE, Browne TA, Lewis AA, Kunz GM, Reitmeier M, Utter RK, Simone M (2022) Threats to online surveys: recognizing, detecting, and preventing survey bots. Soc Work Res 46(4):343–350. https://doi.org/10.1093/swr/svac023

Yarrish C, Groshon L, Mitchell J, Appelbaum A, Klock S, Winternitz T, Friedman-Wheeler D (2019) Finding the signal in the noise minimizing responses from bots and inattentive humans in online research. Behav Ther 42:235

Zhang Z, Zhu S, Mink J, Xiong A, Song L, Wang G (2022) Beyond bot detection: combating fraudulent online survey takers. In: Proceedings of the ACM web conference. ACM, pp 699–709. https://doi.org/10.1145/3485447.3512230. Accessed 19 Nov 2025