Case Studies on the Motivation and Performance of Contributors Who Verify and Maintain In-Flux Tabular Datasets

SHAUN WALLACE, Brown University, USA
ALEXANDRA PAPOUTSAKI*, Pomona College, USA
NEILLY H. TAN*, University of Washington, USA
HUA GUO*, Twitter, USA
JEFF HUANG, Brown University, USA

The life cycle of a peer-produced dataset follows the phases of growth, maturity, and decline. Paying crowdworkers is a proven method to collect and organize information into structured tables. However, these tabular representations may contain inaccuracies due to errors or data changing over time. Thus, the maturation phase of a dataset can benefit from the additional human examination. One method to improve accuracy is to recruit additional paid crowdworkers to verify and correct errors. An alternative method relies on unpaid contributors, collectively editing the dataset during regular use. We describe two case studies to examine different strategies for human verification and maintenance of in-flux tabular datasets. The first case study examines traditional micro-task verification strategies with paid crowdworkers, while the second examines long-term maintenance strategies with unpaid contributions from non-crowdworkers. Two paid verification strategies that produced more accurate corrections at a lower cost per accurate correction were redundant data collection followed by final verification from a trusted crowdworker and allowing crowdworkers to review any data freely. In the unpaid maintenance strategies, contributors provided more accurate corrections when asked to review data matching their interests. This research identifies considerations and future approaches to collectively improving information accuracy and longevity of tabular information.

CCS Concepts: • **Human-centered computing** → **Human computer interaction (HCI)**; **Computer supported cooperative work**; • **Information systems** → **Crowdsourcing**; *Data cleaning*; *Asynchronous editors*; Incomplete data.

Additional Key Words and Phrases: Tabular Data; Data Verification; Data Maintenance; Unpaid Contributions; Crowdsourcing; Peer Production

ACM Reference Format:
Shaun Wallace, Alexandra Papoutsaki, Neilly H. Tan, Hua Guo, and Jeff Huang. 2021. Case Studies on the Motivation and Performance of Contributors Who Verify and Maintain In-Flux Tabular Datasets. *Proc. ACM Hum.-Comput. Interact.* 5, CSCW2, Article 448 (October 2021), 25 pages. https://doi.org/10.1145/3479592

*Work began while author was at Brown University

Authors' addresses: Shaun Wallace, Brown University, USA; Alexandra Papoutsaki, Pomona College, USA; Neilly H. Tan, University of Washington, USA; Hua Guo, Twitter, USA; Jeff Huang, Brown University, USA.

2573-0142/2021/10-ART448 $15.00
https://doi.org/10.1145/3479592

1 INTRODUCTION

There are numerous collective Peer Production and Citizen Science efforts to build datasets, such as FLOSS, Wikis, and Knowledge Graphs. These datasets, powered by human-collaboration, have life cycles where data is collected, matures, and eventually dies [28]. The maturation process is influenced by data that is in flux. This data requires constant updates to maintain its accuracy and utility. A popular but often overlooked interface to collect and improve the accuracy of a tabular dataset is the spreadsheet. A qualitative 3-year study shows spreadsheets are the primary tool to clean, or improve the accuracy of data in professional settings [11]. There are recent successful peer production efforts to collect and maintain tabular data started at the onset of the COVID-19 pandemic, as people making unpaid contributions (i.e., edits) have maintained an in-flux spreadsheet of university campus closures and migrations to online learning [1]. These observations show spreadsheets, used by over 750 million people in Excel alone, are an ideal choice to study the maturity phase of a tabular dataset's life cycle [21].

Our two case studies focus on the motivation and performance of contributors during the maturity phase of a tabular dataset's life cycle compared to the large body of related work focusing on improving unstructured data [28, 34, 72]. Tabular datasets, structured as unique rows and corresponding columns, enable users to engage in structured information exploration and retrieval to augment semantic knowledge bases. [83]. Disseminating tabular information from human knowledge and online information sources is a collective information retrieval and organization task often powered by peer production systems and strategies. However, accurate datasets are difficult to maintain during the maturation phase. Error-prone tabular data [56] can lead to dire consequences, like the austerity measures imposed on Greece after the 2008 financial crisis which were based on a spreadsheet with numerous errors that inaccurately represented the relationship between public debt and GDP growth [26]. It is imperative to verify and maintain tabular datasets' accuracy to enable positive impacts continuously. Not all data is static; to maintain its accuracy as it grows in size and evolves, it is necessary for groups of people to collectively review and edit data.

This paper conducts two different case studies on the maturation phase of in-flux tabular data's life cycle. Previous research has shown that crowdsourcing is a popular low cost and time-efficient method to collect tabular data for the initial collection phase [35]. Without implementing quality control measures, crowdsourcing can yield inaccuracies with negative consequences [82]. One set of strategies we explore to verify and correct tabular data is to pay crowdworkers, while an alternative method includes continuous edits, or maintenance strategies, from unpaid contributors over 2.5 years. We conduct two separate case studies to examine how each method, paid crowdworkers and unpaid contributors, improve crowd-collected tabular datasets' accuracy. The first case study focuses on what we call **verification strategies** and includes five strategies we adapted from literature [5, 29, 35, 47] that rely on posting micro-tasks to recruit paid crowdworkers to edit tabular data quickly. The second case study examines **maintenance strategies**, which include two strategies adapted from literature [74, 77, 80] that rely on waiting for unpaid contributors to continuously visit and make unsolicited and solicited edits to tabular data.

The two case studies grew organically through our attempts to maintain a tabular dataset containing information on Computer Science faculty profiles since 2016 [77]. While Wikipedia often features tabular data on individual pages, these tables contain a relatively small number of rows and columns. Our case studies feature larger tabular datasets consisting of thousands of rows and at least ten columns. Paid crowdworkers collected the initial dataset, and then we hosted it in a custom editable spreadsheet web application. To the best of our knowledge, no published work has studied the maintenance of a tabular dataset by unpaid contributions over multiple years. Instead, our second case study has attracted thousands of visitors who voluntarily corrected

data and added new faculty information without monetary compensation. After observing this naturalistic maintenance by unpaid contributors, we chose to explore how traditional methods using paid crowdworkers would work under similar circumstances. A verification phase using paid crowdworkers was started immediately after collecting a second dataset using paid crowdworkers to improve its accuracy. Equipped with two tabular datasets on the same topic, we analyze the strategies examined within each case study.

This paper identifies the trade-offs of five individual verification strategies and then two maintenance strategies for enabling groups of people to collaboratively verify and maintain in-flux tabular data. We study the accuracy achieved and the payment or time required for human-driven "table augmentation" tasks (i.e., edits): fixing existing data, filling in empty cells, and adding new rows of data [83]. We also investigate the accuracy of different data types that may require varying levels of subjectivity and domain-specific knowledge.

Within the paid verification strategies, we review the cost required for an accurate edit, while within the unpaid longitudinal maintenance strategies, we review the time needed to wait for an accurate edit and whether a few users make the majority of edits (similar to Wikipedia). We also examine if the number of visits, edits, or types of interactions can indicate whether unpaid contributors vandalize data during long-term deployment.

Overall, we find that verification strategies are more accurate at a lower cost when rehiring trusted crowdworkers. On the other hand, groups of unpaid contributors improved data regularly through small contributions with no signs of vandalism. They also excelled at editing subjective data types requiring domain-specific knowledge. The results of these separate case studies create discussions focusing on: (1) selecting appropriate paid or unpaid approaches to improve the accuracy of tabular data, (2) new hybrid approaches blending lessons learned from each case study, and (3) proposing new automated methods to attract unpaid contributions in peer production systems.

Our research independently explores verification and maintenance strategies that enable collaborative editing behaviors. We avoid a direct experimental comparison between paid crowdworkers and unpaid contributors due to the inherent difficulty of acquiring naturalistic editing behavior and the constraints of running parallel longitudinal studies. Instead, by examining two naturalistic case studies on the same type of data, we gain insights that can inform future studies to create a controlled study design for a direct comparison. As part of our contributions, we release the datasets and labels from this work[1] so others can further these efforts.

2 RELATED WORK

Dataset curators, also known as requesters, often recruit paid crowdworkers by posting short, repeatable micro-tasks to quickly collect information for beneficial datasets [15, 36]. For example, crowdworkers validated alt-text tags to improve the usability of social media posts for blind users [69] and collected a dataset to help researchers understand personality disorders [50].

Researchers have studied several scalable techniques to assess and improve the quality of crowd-collected data. While paying crowdworkers is useful to collect data, there are concerns over the resulting accuracy. Reasons for this can include varying motivation [53] and effort levels between crowdworkers [5], malicious crowdworkers [22], and insufficient expertise [71].

2.1 Paid Approaches to Improve Datasets

Paying crowdworkers to verify crowdsourced datasets is a common method to improve their accuracy [63]. In our first case study, we adapt classic crowdsourcing approaches to generate five

[1] Available at http://drafty.cs.brown.edu.

verification strategies for correcting tabular datasets. We adapt Bernstein et al.'s popular Find-Fix-Verify method [5] and Hirth et al.'s Majority Decision [29]. We also introduce variations of them inspired by Marcus' tactics for knowledge-specific tasks [47]. Balancing the speed and cost of recruitment in these strategies is difficult. Huang and Bigham developed the Ignition Framework combining on-demand recruiting and the retainer model to balance recruiting workers' cost and immediacy [32]. While it proved successful, this method would require large sums of money to pay crowdworkers to improve an in-flux tabular dataset accuracy continuously.

Past work focused on methods to recruit paid crowdworkers to solve specialized tasks. Kittur et al. [37] increased the quality in solving complex problems by using simpler micro-tasks. Likewise, in Turkomatic, crowdworkers simplify complicated steps recursively, then, other crowdworkers verify solutions and form an answer [41]. These approaches require extra time for requesters to simplify tasks and manage multiple crowdworkers. Previous efforts also tried to match crowdworkers to micro-tasks matching their skills and expertise [51]. Our work extends these ideas by allowing requesters to recruit specific crowdworkers to perform verification tasks.

2.2 Unpaid Approaches to Improve Datasets

A second approach to improve the quality of crowdsourced datasets is to rely on unpaid contributors' interests and expertise to ensure accurate and continuous corrections. Xu and Maitland [81] employed similar concepts while studying users' participatory data maintenance in field trials with urban refugees. Quattrone et al. [62] studied geographic maintenance practices within OpenStreetMap, where users freely update spatial information. Additional past work has relied on user-interest in information to attract and solicit edits [77] within a tabular dataset. While this short-term study also identified domain-specific challenges in data maintenance, it did not capture edits where dataset change naturally over time. Our work builds on these past ideas by studying visitors' interactions with a tabular dataset while acting as unpaid contributors who edit data over time.

Past work from recommender systems shows building user interest profiles using implicit feedback from user interactions can successfully target users [84]. Other work has focused on unpaid approaches to targeting users. For example in peer production systems, such as SuggestBot [12] and Wikitasks [40], utilize user interactions and edit histories to suggest edits and assist in task design. In contrast to Wikipedia [64], where most content is long-form articles and some small tables, our work focuses on collaborators editing in-flux tabular data consisting of thousands of rows presented in spreadsheets.

2.3 Improving Domain-Specific Knowledge

Maintaining data over long periods poses unique challenges for soliciting edits, especially for domain-specific data. A simple approach is to continuously post paid micro-tasks for these knowledge-intensive tasks [23, 55, 71] or use custom systems [24]. For example, in Crowdfill [59], crowdworkers freely edit a spreadsheet and up/down-vote possible data points. This closed system gives crowdworkers freedom but does not allow requesters to employ verification tasks freely. While paying crowdworkers is an effective solution for short-term verification, it can become less practical over time due to financial constraints, time, and data integrity [48].

2.4 Motivating Paid and Unpaid Contributors

Self-determination theory defines two types of motivation: intrinsic and extrinsic [14]. Intrinsic motivation refers to an individual's inherent desire to participate in an activity, whereas extrinsic motivation is influenced by conditional rewards (e.g., money) [2, 68]. Our work does not seek to

directly answer what types of motivation influence paid crowdworkers and unpaid contributors. However, we want to acknowledge the related work covering these motivations.

Historically, payment per task can extrinsically motivate crowdworkers [30]. Also, meta-incentives can augment extrinsic motivation for future payments. For example, rejecting low-quality work can negatively affect a crowdworker's approval rate, thus impacting their ability to complete future work [46]. In our case study on verification strategies, requesters had to balance these constraints and determine appropriate compensation schemes per task to ensure quality work.

People may be intrinsically motivated to contribute their time toward causes they are interested in [51, 52]. Our case study on maintenance strategies relies on visitors to a tabular dataset making unpaid edits. We posit they might participate because of an interest in Computer Science. This reliance on user interest has proven successful in maintaining a popular resource in the CS community, CSRankings [3]. To the best of our knowledge, there is no equivalent system relying on paid crowdworkers for edits.

Previous research has explored hybrid models where users are subjected to intrinsic and extrinsic motivators. In prior work by Flores-Saviaga et al., unpaid contributors were more effective at open-ended tasks, such as original content creation, while paid crowdworkers were more effective at completing simpler tasks following strict guidelines [8, 20]. As a result, they suggest a hybrid approach: leveraging volunteer work for original content, and employing crowdsourced work to structure and prepare content for real-world use. Our research provides new insights about potential hybrid models by studying the results of two separate case studies where paid and unpaid users completing similar tasks.

3 OVERVIEW OF CASE STUDIES

This work presents two case studies focusing on collaborative efforts to improve the accuracy of in-flux tabular datasets, one of the most fundamental ways of organizing information [21]. This section explains the structure of the collected data and the study design considerations for each case study. Our Institutional Review Board classified the procedures as exempt from review. The requesters, paid crowdworkers, and unpaid contributors were informed and consented to their data being used for research.

3.1 Structure of Tabular Datasets

Tabular datasets consist of data organized in tables with horizontal rows and vertical columns [75]. Each row can represent a unique entity with the same number of cells. Each cell corresponds to the intersection of a specific row and column. Columns correspond to a particular property characterizing data per row. Tabular datasets exist in various formats (e.g., comma- and tab-separated values files) and can be edited using dedicated spreadsheet programs like Microsoft Excel or Google Sheets. Our work focuses on strategies to enable groups of people to collaboratively build high-quality relational data, also known as "relational tables" [83].

In the case study on maintenance strategies, unpaid contributors used a publicly-accessible web-based editable spreadsheet interface. In the case study on verification strategies, paid crowdworkers used Google Sheets. Following prior recommendations [13], all interfaces used built-in inputs to validate data. Each case study used a distinct instance of a tabular dataset of Computer Science faculty academic profiles from top programs in the United States and Canada. Each academic profile is a row in a spreadsheet. Each column corresponds to their affiliated university, the year they joined as faculty, rank, subfield area of expertise, and where they received their Bachelors, Masters, and Ph.D. degrees, and sources used to gather the information. During both case studies, the profiles also featured gender at the request of researchers who wanted to analyze hiring trends.

Our tabular context closely resembles the collaborative community efforts to build and maintain in-flux tabular data, such as the COVID-19 dataset mentioned in the Introduction [1]. Our case studies focus on the maturity phase of a tabular dataset and extend prior work on tabular data from Wikipedia and Wikidata [6, 19]. For example, while our tabular datasets focus on a semantically cohesive concept, Computer Science professors, and span thousands of rows and more than ten columns, they share the same standard tabular structure with tables found in Wikipedia. As the size of the tables and tabular data within Wikipedia grow and there are existing tools that can automatically translate tabular data to Wikidata [54, 73], there is potential for our research to generalize to wider audiences using these tools and platforms.

3.2 Study Design Considerations

The case study on maintenance strategies spanned years and grew organically in a dataset we hosted that naturally attracted visitors who contributed to it without any payment. Therefore, we made specific design decisions to study verification strategies in a naturalistic setting. This section covers how the data was initially collected and the study design decisions we made to study realistic editing behaviors.

For each case study, undergraduate and graduate students from two separate human-computer interaction seminars acted as requesters, as part of a graded class assignment, to employ crowdworkers to collect a different faculty dataset following the "Classic Micro" data collection strategy [57]. In this strategy, a task translates to finding all the information on a specific faculty member. Twenty students followed these methods in 2015 to recruit crowdworkers to collect the initial dataset for the maintenance strategies relying on unpaid contributors. In 2018, twelve students recruited crowdworkers to collect a new dataset for the case study of verification strategies. These same twelve students recruited crowdworkers again to complete the verification strategies. We did not reuse the dataset collected in 2015 for the case study of verification strategies; over the three years that elapsed, the data continuously evolved due to promotions and new faculty hires, among other naturally occurring events. A newly collected dataset in 2018 would contain errors due to mistakes made by crowdworkers during the collection phase or inconsistencies in online sources.

To avoid biases due to our experience with crowdsourcing and promote a naturalistic approach similar to those observed in the maintenance strategies, we did not conduct the verification strategies ourselves. Instead, similar to Papoutsaki et al. [57], students within a class acted as requesters employing paid crowdworkers within the verification strategies. All requesters were given the same budget and two weeks as a time constraint for each verification strategy. Although each requester was assigned the same five verification strategies, they were free to experiment with payment combinations regarding pay-per-task, bonuses, and communication techniques with crowdworkers.

4 CASE STUDY I: PAID VERIFICATION STRATEGIES

We conducted an exploratory case study on paid data verification strategies for in-flux tabular datasets by observing requesters employing paid crowdworkers in a naturalistic scenario. In contrast to prior work on data verification [5, 29, 32], our work features a real-world scenario where requesters have to balance the natural constraints of time and money to achieve results without the aid of a novel system.

4.1 The Verification Strategies

We defined five verification strategies by adapting popular micro-task verification strategies from literature to edit tabular datasets. The strategies are summarized below and in Figure 1.

(1) *Find-Fix-Verify*: A popular strategy introduced by Bernstein et al. [5]. Three sets of unique crowdworkers perform each task: the first set identifies errors, the second set fixes errors, and a final set verifies the information's accuracy.
(2) *Find-Fix*: The first component of *Find-Fix-Verify*. Two sets of unique crowdworkers perform each task: the first set identifies errors and the second set fixes errors.
(3) *Find+Fix*: The second component of *Find-Fix-Verify*. A single crowdworker is required to find and fix inaccurate data.
(4) *Majority Rule*: Similar to *Majority Decision* proposed by Hirth et al. [29], the most common response per data point was deemed correct. Unique sets of crowdworkers redundantly collect sets of data until two or more sets are in agreement.
(5) *Expert Rule*: A variation of *Majority Rule*, was inspired by Marcus' [47] tactic of enlisting a trusted crowdworker as an "expert" to review tasks performed by others to ensure their accuracy. Unique sets of crowdworkers redundantly collect duplicate sets of data. A third crowdworker then compares the multiple sets to determine the correct data.

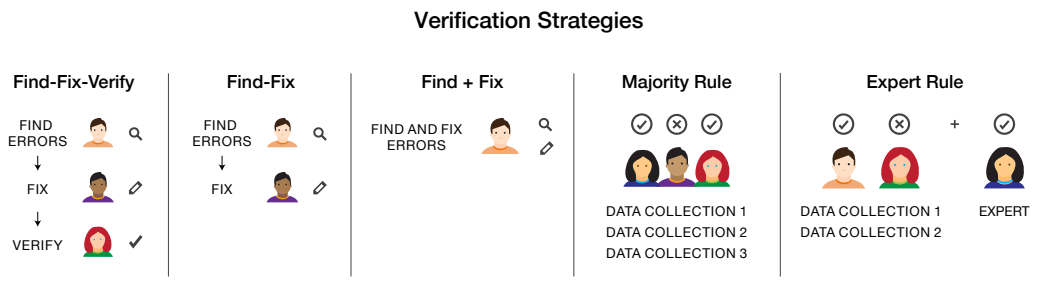

Fig. 1. The five verification strategies outlined above require paid crowdworkers for each verification step. For example, *Find-Fix-Verify* and *Find-Fix* require a unique crowdworker for each step, whereas a single crowdworker performs *Find+Fix*. Both *Majority Rule* and *Expert Rule* rely on redundant data collection, while *Expert Rule* relies on a trusted crowdworker to review data and break ties.

In this paradigm, trusted crowdworkers might not possess domain expertise, but might have previously completed tasks for the requester and are thus deemed an expert. The ability to re-recruit trusted crowdworkers as experts follows Daniel et al.'s [13] recommendation for requesters to develop long-term relationships with crowdworkers. In our study, requesters employed their methods to recruit trusted crowdworkers on a task-by-task basis per strategy.

We limited the number of verification strategies to five following the advice from Wiggins et al. [79] who found a negative correlation between the number of verification techniques used and money paid after analyzing a collection of citizen science experiments.

4.2 Methods: Verification Strategies

In Spring 2018, twelve students from a human-computer interaction seminar acted as requesters as part of a graded assignment. Each student (requester) employed paid crowdworkers to verify a crowd-collected tabular dataset over two weeks. Requesters used all five verification strategies and were randomly assigned one university per strategy. We removed three requesters' data from our analysis for not following the study procedure.

4.2.1 Requesters balancing crowdworker compensation, quality, and time. Requesters had to balance their budget for verification tasks. Some of these tasks were repetitive, such as those in *Majority Rule* or *Expert Rule*, and required duplicate data for verification. Requesters naturally accounted

for this strict 2-week constraint, echoed by Faridani et al.'s [18] recommendation, by balancing compensation against desired completion time. Therefore, this makes it difficult to forecast costs for tasks focusing on large datasets requiring repeatable tasks. For example, Mason and Watts [48] discovered that an increase in payment per task results in increased quantity of work, but not necessarily quality.

4.2.2 Initial steps and recruitment of crowdworkers. First, requesters read three seminal works on crowdsourcing [5, 17, 35] and watched a talk by Marcus on working with crowdworkers [47]. Requesters were randomly assigned five universities to collect and then verify data for using paid crowdworkers and a unique verification strategy per university. We checked the number of professors that requesters would need to collect and then verify, ensuring a balanced sample. Each requester received $50 in credit to use on Amazon Mechanical Turk (AMT), informed by the difficulty of tasks and recommendations from [57]. The experiment consists of three phases: testing, data collection, and data verification. Requesters were required to spend $5 in total for testing. During testing, they could experiment with AMT and develop, test, and review each strategy's successes. Then, the requesters had $4 per university to spend on AMT to collect data. After that, they had $5 to spend for data verification on AMT as well. During the verification phase, requesters could experiment with payment structures, but could not use preset qualifications.

4.2.3 How edits are made. Requesters hosted each dataset per university on a separate instance of Google Sheets to allow crowdworkers to only edit data for the assigned verification strategy. Verification was only performed after the collection phase complete. We chose not to direct crowdworkers to the existing web platform used to study maintenance strategies to ensure that unpaid contributors and paid crowdworkers could not access each others' datasets.

Verification Strategy	Total Labeled Edits	Total Edits	Total Payment	Payment per Edit	Payment per Accurate Edit	Payment Increase Accurate Edit
Find-Fix-Verify	190	298	$67.97	$0.23	$0.32	42%
Find-Fix	200	472	$73.19	$0.16	$0.27	71%
Find+Fix	197	311	$44.74	$0.14	$0.23	60%
Majority Rule	222	674	$44.77	$0.07	$0.11	61%
Expert Rule	207	529	$54.63	$0.10	$0.15	48%
All	1,016	2,284	$285.30	$0.13	$0.19	56%

Table 1. A summary of edits made by paid crowdworkers in the case study of verification strategies. It includes the number of edits we manually labeled as correct/incorrect out of all edits, the total amount of money spent on verification strategies, the payment for each edit, and the increase required to obtain an accurate edit. *Expert Rule* required the least amount of additional money to generate accurate edits.

4.3 Results: Verification Strategies

Requesters spent a total of $285.30 to verify data, generating 2,284 edits at the average cost of $0.13 per edit. The total costs include payment-per-task and bonuses. To compute each strategy's accuracy, we first used stratified sampling to select edits to label as correct or incorrect. We compared each edit's value when it was made with faculty web pages, LinkedIn profiles, and resumes. We labeled 1,016 edits by hand, shown in Table 1. The overall accuracy for verification strategies (64%) reported in Table 2 is computed by summing the total correct edits per verification strategy over the sum of the total edits. To understand the cost needed to generate a correct edit, we compute cost-per-correct-edit by dividing the cost-per-edit over the total number of correct edits, as reported

in Table 1. The average cost per correct edit, $0.19, is 56% higher than the cost to generate an edit. The higher the accuracy, the more money goes towards generating accurate data.

4.3.1 Trade-offs between Verification Strategies. Each strategy has advantages and disadvantages. *Majority Rule* and *Expert Rule* generated accurate data at less cost compared to strategies adapted from *Find-Fix-Verify* (Table 1). This shows that collecting duplicate data is a cost-effective method. *Find-Fix-Verify* has the highest overall accuracy across all strategies at 70%, and *Expert Rule* is second, at 69%. Both strategies recruit an additional crowdworker to verify or break ties. This extra step increases costs but generally leads to better accuracy overall and per column. This finding points to a future question on the number or quality of crowdworkers needed to review information until it is correct.

Accurate strategies allow requesters to generate correct data and control costs. For example, *Find-Fix*, the least accurate strategy, required a 71% increase in cost to generate correct edits. *Find-Fix-Verify* increased costs because it recruits an additional crowdworker, but yields a 17% increase in accuracy. The best strategy to control and decrease costs is *Expert Rule*. It achieves this at less than half the cost-per-edit than *Find-Fix-Verify*. These findings support the idea that hiring additional paid crowdworkers to verify data will improve its accuracy while reducing overall costs for maturing a tabular dataset.

		—— *Less subjective data types (columns)* ⟶					
Verification Strategy	**Overall Accuracy**	**Sub-field**	**Join Year**	**Rank**	**Masters**	**Bachelors**	**PhD**
Find-Fix-Verify	53%	56%	75%	72%	70%	91%	70%
Find-Fix	36%	52%	80%	63%	68%	81%	60%
Find+Fix	38%	62%	81%	61%	67%	83%	62%
Majority Rule	47%	59%	66%	57%	66%	68%	61%
Expert Rule	65%	59%	79%	70%	66%	78%	69%
All	48%	58%	75%	64%	67%	80%	64%

Table 2. Overall accuracy per strategy per column in the case study of verification strategies. *Expert Rule* has the highest overall accuracy when accounting for edits across all columns. Subfield, the most subjective data type, has the lowest accuracy per data type. Identifying a professor's subfield requires domain-specific knowledge and can be difficult for crowdworkers to interpret correctly. Accuracy is the average of the total correct edits over the total edits across all strategies.

4.3.2 Differences across Data Types. Each data type/column within tabular data has different properties that requesters had to consider. For example, a professor's subfield may require crowdworkers to have domain-specific knowledge to understand the differences between research areas. *Expert Rule* has the highest accuracy for subfield at 65%, as shown in Table 2. *Expert Rule* and *Find-Fix-Verify* outperform their simpler variations. Notably, *Expert Rule* produced higher levels of accuracy across every data type compared to *Majority Rule*. This observation shows that trusted crowdworkers might possess the requisite expertise and effort needed to correct information that is more difficult to find and understand. Results show that data types requiring domain-specific knowledge can benefit from multiple collection efforts, explaining why *Expert Rule* and *Majority Rule* outperform other strategies. In contrast, identifying where a professor received their Ph.D. or their rank are subjectively easier tasks, with an accuracy of 80% and 75%, respectively. Rank has three possible values: Assistant, Associate, or Full, while a professor's Ph.D. is often easy to find. Results show that easy-to-collect fields benefit from individuals reviewing the data while collecting duplicates

could suffer from potential noise. Overall, we identify that different strategies are best suited for different data types. Future requesters should select the best verification strategy based on the occurrence or importance of a tabular dataset's data types.

4.3.3 Filling in Empty Cells & Adding New Rows. Filling in empty cells or adding missing rows is an essential step to creating an accurate dataset. Collecting duplicates is a quick and cheap method to generate more data, but duplicates may introduce noise. As shown in Table 3, our data shows that *Expert Rule* is useful for filling in empty cells, possibly because an additional crowdworker can sort through this noise selecting the correct value. We observed a drawback of relying on paid crowdworkers when adding new rows. They often incorrectly added rows for non-tenure-track positions (Lecturer, Adjunct, Staff, or Professor of Practice) or professors with an appointment in a non-Computer Science department (e.g., Engineering, Media, Computational Biology).

Verification Strategy	**Overall Accuracy**	*Filling in Empty Cells*			*Adding New Rows*		
		N	**Labeled**	**Accuracy**	**N**	**Labeled**	**Accuracy**
Find-Fix-Verify	53%	118	91	66%	90	57	77%
Find-Fix	52%	156	109	63%	20	13	54%
Find+Fix	62%	101	62	56%	44	35	80%
Majority Rule	47%	222	94	55%	98	65	68%
Expert Rule	65%	239	111	72%	24	19	63%
All	48%	836	467	63%	276	189	71%

Table 3. Overall accuracy across verification strategies, and for filling in empty cells, and adding new rows. *Find+Fix* has the highest accuracy for newly-added rows, while *Expert Rule* was the most accurate for making edits to empty cells. Accuracy is the average of the total correct edits over the total edits across all strategies.

4.3.4 Paying for Correct Edits. Managing a strict budget to build an accurate tabular dataset can be a difficult task. Using strategies with high accuracy can ensure requesters do not need to post additional micro-tasks to acquire accurate data. Overall, in the verification strategies, requesters spent $0.13 per edit to recruit crowdworkers, as shown in Table 1. These numbers increase when accounting for accuracy: requesters paid crowdworkers $0.19 per correct edit when posting micro-tasks. This 56% increase in the money needed to acquire a correct edit can make it difficult to predict the exact cost to verify an accurate dataset. To keep costs predictable, we recommend using a strategy such as *Find-Fix-Verify* or *Expert Rule* that employs a trusted worker to perform a final verification step.

4.4 Takeaways: Verification Strategies

The closest dataset to ours [57] includes faculty profiles that were collected and not verified. In our work, we only focus on the accuracy of the edits and not of the entire dataset. Thus, the accuracy we report (64%) is not directly comparable with the overall accuracy (74%) of the entire dataset presented in [57].

Contrary to similar prior work [57], we report accuracy per data type. For the verification strategies, these accuracies are consistently low. A possible reason for this is that data verification tasks might be more difficult than data collection tasks for paid crowdworkers. Verification tasks might involve correcting data that is more difficult to find and interpret because the initial data collection efforts were unsuccessful. Thus, these complex verification tasks do not resemble the more straightforward tasks paid crowdworkers often excel at [8, 20].

Verification strategies requiring an extra crowdworker to perform a final verification step, such as *Find-Fix-Verify* and *Expert Rule*, are more accurate. They perform better when correcting data that require domain-specific knowledge, confirming past findings of a similar relationship when using paid crowdworkers [23, 71]. Therefore, if a tabular dataset contains a large amount of subjective data, we recommend that requesters use verification strategies requiring an additional trusted worker.

Across the five strategies, *Expert Rule* controls costs the best. Its cost-per-edit is less than half than the second-best strategy *Find-Fix-Verify*. *Expert Rule* has an initial low-cost redundant data collection step, followed by a beneficial final verification from a trusted crowdworker. This mirrors Bernstein et al. experience when developing their implementation of *Find-Fix-Verify* to edit Word documents [5].

Expert Rule proved to be the most effective verification strategy to find and correct empty cells. Crowdworkers specifically recruited to perform this final verification have often seen the dataset before, making it easier for them to navigate and make edits. Their payment is also often increased per task, helping to explain that in addition to being a repeat trusted crowdworker, the higher payment can garner higher quality work [30].

In our verification strategies, requesters first generate the names of all the professors per university at a given point in time. If this initial step produced an incorrect list, this would cause the requester to run additional tasks. A requester could reduce these additional tasks if they run a task to verify this initial list of rows. Therefore, we recommend that future requesters using these verification strategies integrate this pre-collection task to verify they have the correct rows per entity to reduce the risk of running additional unnecessary tasks. As a parallel to managing data in a Knowledge Graph like Wikidata, this recommendation ensures data curators verify they had the correct number of nodes before paying crowdworkers to add additional metadata per node.

5 CASE STUDY II: UNPAID MAINTENANCE STRATEGIES

The second case study on continuous unpaid data maintenance strategies is a longitudinal observation of unpaid contributors visiting and editing in-flux tabular dataset. This "in the wild" study features two perpetual data maintenance strategies where unpaid contributors make either "unsolicited" or "solicited" edits over two and a half years.

5.1 The Maintenance Strategies

This case study grew organically from our multi-year effort to host a tabular dataset on Computer Science faculty where the data stagnates and needs additional edits. Wikipedia has shown that people freely visiting unstructured data are capable of successful maintenance [64]. Building off this idea, we present two continuous maintenance strategies to improve tabular datasets' accuracy, as shown in Figure 2. In one strategy, unpaid contributors maintain data by making "unsolicited" edits over time as they visit the online tabular dataset. In the second maintenance strategy, unpaid contributors maintain data when the platform hosting the tabular dataset "solicits" them to edit data matching their interests. The system analyzes their prior interactions with the tabular dataset to derive these interests. A unique aspect of these maintenance strategies is that the dataset curator waits for visitors to edit data over time. Hence, instead of paying crowdworkers to make edits immediately, these maintenance strategies can run perpetually by relying on a stream of edits over time by unpaid contributors who visit the tabular dataset in the wild.

5.1.1 Relying on Unpaid Contributors. It is difficult for a curator to know when their dataset is out-of-date or inaccurate and immediately pay a crowdworker to correct it. Instead, our maintenance strategies rely on unpaid contributors visiting over time to perpetually review and edit data. We feel

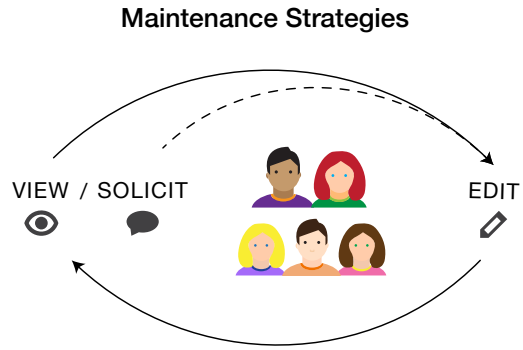

Fig. 2. Maintenance strategies feature a continuous workflow allowing visitors to the tabular dataset to edit data repeatedly. Visitors, acting as unpaid contributors, can freely edit data of their choice or be solicited by the system to edit data that match their interests.

this is necessary because some tabular datasets, such as ours, are not static. The data can change and thus requires continuous review. Data could change because:

(1) The initial information collected could be incorrect, and therefore it needs to be changed.
(2) The initial information was correct, and then someone modified it incorrectly.
(3) The data needs to be updated because it has changed. For example, a professor could change universities or change their subfield area of expertise.

5.2 Methods: Maintenance Strategies

Starting in 2016, we conducted a case study on maintenance strategies in the wild using a publicly-available web application seeded with a crowd-collected dataset of 50,000 values from over 3,600 faculty profiles. We observed this human-centric approach of data maintenance for over two and a half years. We stopped running the case study when the web application hosting the dataset received a significant update in early 2019.

5.2.1 Initial steps. The maintenance strategies rely on waiting for unpaid contributors to edit and maintain the dataset. In our pilot studies on maintenance strategies for in-flux tabular datasets, we found three recommended attributes of a tabular dataset that made it easier to attract visitors and study their contributions.:

(1) Each row of the dataset must remain valid for extended periods. For example, a professor can stay within academia for long periods. However, a faculty job posting could quickly become irrelevant once the position is filled.
(2) Some columns within the tabular dataset should change over time. For example, a professor could change universities. This type of data presents more opportunities for edits.
(3) The tabular dataset needs enough data to attract visitors to make contributions. If a dataset is missing too much information, users may not feel motivated to participate in something that feels neglected.

5.2.2 Recruitment of editors. Attracting interested visitors to make unpaid edits is necessary to study this continuous maintenance approach. When the system was initially seeded, we made posts across various CS forums and websites (Reddit CompSci, Hacker News, and TheGradCafe) to inform and attract an initial user base with related interests. 'LabintheWild' use similar strategies to attract users through social media platforms [65]. An example title used on posts was "Records

of 3,600 computer science professors at 70 top universities (US/Canada) help us keep it up to date!" The goal was to appeal to users interested in Computer Science who may be:

(1) a professor listed in the dataset,
(2) a prospective student looking for an advisor,
(3) a friend, colleague, or family member of a CS professor,
(4) someone (e.g., in administration) who might be interested in running analysis on trends in Computer Science, or
(5) someone who cares about adding to public information.

Each post advertising the dataset contained the text:

> Wanted to share a computer science resource a couple of us in the Brown University Human-Computer Interaction group have put together. It is a crowd-editable spreadsheet of data of approximately 3,600 computer science professors. For example, where they got their degrees, subfield of expertise, their join year and rank, etc... It might be useful if you're applying to Ph.D. programs or faculty positions, seeking external collaborators, or just to better understand hiring trends in CS departments.

We only made these initial posts. All subsequent traffic to the dataset was generated through organic search traffic or other means we did not control. While prior work shows that using social norms can motivate unpaid contributions; we chose not to employ this method [8]. Using such an approach may introduce additional factors affecting users' motivations within our naturalistic and longitudinal case study. Our goal was to assess the maintenance strategies and not to explore how best to build or attract a continuous flow of users.

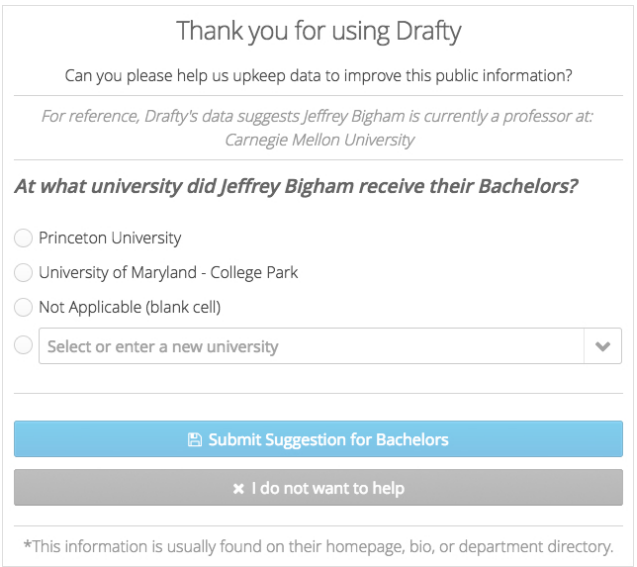

Fig. 3. This interface is the edit cell window from the Drafty web application that appears when Drafty solicits unpaid contributors to edit data based on their interests within the maintenance strategies. They can confirm the data does not exist, submit a correction, or exit to return to the spreadsheet interface.

5.2.3 How edits are made. The two maintenance strategies allow users to edit the data freely at any time without the need to create an account within the system. The system tracks users

anonymously. However, the system presents users with a modal dialog informing them how to make edits on their first visit. Users can freely make *unsolicited* edits at any time by double-clicking a cell in the spreadsheet interface. Users can select a new value from previous edits to that cell, a predefined list of possible values, or freely enter text. The system does not require a special markup language for edits like Wikipedia. Specific data or features are not protected or semi-protected like in Wikipedia or Wikidata. The platform can also *solicit* edits from users by prompting them with a modal dialog request to fix a specific data value, as seen in Figure 3. The application solicits a user to review a row of data matching their interests. The application infers interest by creating a "user interest profile" per user by tracking their interactions (i.e., search, sort, click, edit) within the spreadsheet and computing a relevance metric, like Wallace et al. [77]. The application displays the most recent edit as the correct value per cell in the interface. This is also how Google Sheets handles multiple possible values per cell in our verification strategies.

5.3 Results: Maintenance Strategies

The maintenance strategies were run over 1,025 days from May 26, 2016, to March 3, 2019. Visitors provided 2,651 edits at a rate of 2.6 edits per day. To compute the accuracy per edit, we labeled 1,020 edits by hand as correct or incorrect using stratified sampling following the same procedures with the case study on verification strategies. The findings are summarized in Table 4. We observed a common mistake where unsolicited visitors incorrectly edited a professor's Ph.D., as the university where their Bachelor's or Master's degree. We also observed the same error made by paid crowdworkers in the case study of verification strategies.

Maintenance Strategy	Total Labeled Edits	Total Edits	Total Time	Time per Edit	Time per Accurate Edit	Time Increase per Accurate Edit
Unsolicited	958	2,566	1,025 days	9.6 hours	10.8 hours	13%
Solicited	62	85	1,025 days	12 days	13 days	5%
All	1,020	2,651	1,025 days	9.3 hours	10.4 hours	12%

Table 4. A summary of edits made by unpaid contributors (maintenance strategies), including the number of edits we manually labeled as correct/incorrect, the total time spent waiting for edits. Soliciting unpaid contributors to edit data they are already interested in is the most efficient method to improve a dataset's accuracy, at a 5% increase in time to wait for an accurate edit.

The overall accuracy for maintenance strategies is 89%, as seen in Table 5. Accuracy is the sum of the number of correct edits over the sum of the total edits across both maintenance strategies. The time between correct edits represents how frequently an unpaid contributor submits an accurate edit and is computed by dividing the number of days or hours per edit over the total number of correct edits (Table 4). In the maintenance strategies, the time between correct edits, 10.4 hours, is 12% higher than the time needed to generate an edit. This metric is comparable to a similar study by [77], where their users made 592 edits in a similarly structured dataset over 214 days at 75% accuracy. Their users generated a correct edit every 11.5 hours.

Our case study's extended length, combined with frequent visitors, could explain why our maintenance strategies have higher accuracy than studies with similar levels of edits per hour. Our case study's strategies produce minimal negative contributions (e.g., incorrect edits), supporting the idea that long-term maintenance is possible in public datasets.

5.3.1 Differences Across Data Types. Previous work shows users with expertise in specialized tasks or interest in the data perform better [71, 77]. Unpaid contributors may have an interest in or prior

		⟵ *Less subjective data types* ⟶					
Maintenance Strategy	**Overall Accuracy**	**Sub-field**	**Join Year**	**Rank**	**Masters**	**Bachelors**	**PhD**
Unsolicited	89%	91%	82%	90%	79%	73%	88%
Solicited	95%	100%	75%	67%	100%	80%	
All	89%	91%	82%	89%	82%	73%	88%

Table 5. Accuracy overall and per strategy per column. *Unsolicited* unpaid contributors excelled at correcting empty cells; this might be because of their prior knowledge of CS professors. Accuracy is the average of the total correct edits over the total edits across all strategies.

knowledge of the data, leading to higher accuracy when editing challenging information. This can help explain the varying levels of accuracy across the data types seen in Table 5 and suggests their domain-specific knowledge helps them accurately identify Rank and Subfield from personal web pages, publications, or their prior knowledge. In contrast, columns with less subjective data, such as Bachelors and Masters degrees, have lower accuracy. These data types are more difficult to find upon review: professors do not always list their degrees on their websites or other sources, whereas their Ph.D. is prominently displayed. Over our longitudinal study, 33% of participants edited at least one value for Rank or Subfield. While these data types require domain-specific knowledge, they can also change over time. This initial result points to the potential for unpaid contributors interested in the data to maintain these types of data over time. Overall, the maintenance strategies benefit from waiting for someone with domain-specific or pre-existing knowledge to assess those more complex data types.

5.3.2 Filling in Empty Cells & Adding New Rows. Table 6 shows that the accuracy for editing empty cells (93%) is higher than the accuracy for edits to cells with existing data. This observation might be due to unpaid contributors having pre-existing knowledge of particular rows in a dataset. For example, they can quickly correct an empty subfield because they already know a professor's area of expertise from reading their research papers. Prior knowledge can make it easy to add a new professor to a university because they know that professor. Newly-added rows are 86% accurate, a level of accuracy similar to unsolicited users correcting existing data.

When users decide to contribute an edit, they could edit existing data, fill in empty cells, or add multiple new cells by adding a new row. Each of these requires varying levels of effort to complete. The following results count the number of times each user edits existing data, fills in empty cells, or the number of new cells they created when adding new rows of data. During our case study, 33% of unpaid contributors filled in mostly empty cells, while 42% edited cells with pre-existing data. The remaining 25% of unpaid contributors edited the same number of empty and non-empty cells. In a similar analysis, we compared the percentage of unpaid contributors filling in more empty cells than creating new cells by adding new rows. In this comparison, 42% of unpaid contributors filled in mostly empty cells, while 6% created new cells when adding new rows. The remaining 52% of unpaid contributors created the same number of new data points when filling in empty cells or adding new rows. This analysis shows unpaid contributors were active filling in empty cells, but few primarily added new rows of data. Adding new rows of data requires more effort; thus, our user base often made edits requiring less effort. As a tabular dataset matures, its balance of empty to non-empty cells will change. Thus, it is essential to view these findings within the context of a tabular dataset that is the beginning of its maturity phase.

Maintenance Strategy	Overall Accuracy	*Filling in Empty Cells* N	Labeled	Accuracy	*Adding New Rows* N	Labeled	Accuracy
Unsolicited	89%	1,219	524	93%	224	99	86%
Solicited	95%	35	20	87%	–*	–*	–*
All	89%	1,254	544	93%	224	99	86%

Table 6. Overall accuracy across maintenance strategies and for filling in empty cells, and adding new rows. *Unsolicited* unpaid contributors excelled at correcting empty cells. Their prior knowledge of CS professors possibly contributes to these high levels of accuracy. Accuracy is the average of the total correct edits over the total edits across all strategies. *Solicited users in the case study of maintenance strategies were not asked to add new rows.

5.3.3 Soliciting Users to Fix Data. Previous studies show that asking users to fix data they are interested in leads to more accurate edits [77]. Our web application solicited visitors 1,018 times to review and correct data matching their interests. They submitted 85 edits, with an accuracy of 95% (Table 5). In our study, soliciting visitors to review and fix data matching their interests leads to a 7% increase in accuracy compared to visitors making normal unsolicited edits. A two-tailed t-test of unequal variances shows this increase was statistically significant, $t(79) = -2.2$, $p = 0.03$. While soliciting visitors to fix data matching their interests proved effective, motivating visitors to complete these edits proved difficult. Only 7.5% of solicitations resulted in submitted edits. A related short-term study showed a similar rate of 8.8% of solicitations resulted in submitted edits [77]. These continued low submission rates across our long-term study show this is an area for future research. Later, we discuss future methods to increase these rates and provide more edits in a shorter amount of time.

5.3.4 Waiting for Correct Edits. This work shows the potential for continuous maintenance strategies to have consistently high accuracy levels, leading to predictable wait times to generate correct edits. While unsolicited, unpaid contributors submitted an edit every 9.3 hours, each correct edit was submitted every 10.4 hours. This 12% increase in the time to wait for a correct edit, as shown in Table 4, shows that consistently accurate edits can benefit continuous maintenance efforts. These findings show that if a dataset curator has the time to invest, maintenance strategies can generate an accurate dataset. Compared to prior work where users might be motivated by initial posts marketing a dataset or through gamified rewards [66], our results indicate a dataset's maturity phase can be extended by relying on user's interested in the data.

5.3.5 Detecting Vandalism and Unpaid Contributor's Edits, Visits, & Interactions. This section reviews unpaid contributor's edits (frequency and content), number of visits, and total interactions (edits, clicks, searches, and sorts) to determine if vandalism occurred during our long-term deployment of the maintenance strategies. Previous research developed models to predict vandalism from user data such as edits and visits to open knowledge graphs in Wikidata [25, 74].

Our maintenance strategies use a custom web application to track interactions with tabular data in spreadsheets. In our study, 82% of unpaid contributors only submitted correct edits. While the accuracy of edits from only 15% of unpaid contributors were below the average accuracy of 89%, as seen in Table 5. An unpaid contributor's total number of edits did not correlate with their accuracy ($r = 0.15$). These results show that the maintenance strategies do not have to rely on power users (i.e., those making the bulk of edits) to create a mature accurate tabular dataset. This result runs contrary to results from Wikipedia, where power users primarily maintain its unstructured data [28, 43, 64].

We did not observe unpaid contributors deleting data or entering values that would be considered inappropriate or incorrect. Errors most likely resulted from incorrect interpretations of data, such as a professor's Rank.

Our multi-year case study allows us to observe the frequency of edits over time compared to short-term work [59, 77]. Excluding the initial months when we made social media posts advertising the dataset, the months with the highest edits were November and December. The months with the lowest number of edits were June and August. November and December align with graduate school and job application periods, while the summer months are not overly active except for faculty members officially starting new positions. This observation shows further evidence of how the maintenance strategies can rely on the interests of unpaid visitors to extend the maturity phase by providing a valuable data source for a community.

An unpaid contributor's total number of visits did not correlate with their accuracy ($r = -0.11$). Also, 97% of unpaid contributors had more than 1 visit. This finding shows having multiple visits and making accurate edits aligns with results from similar work [25, 77]. An unpaid contributor's total number of interactions did not correlate with their accuracy ($r = -0.13$). This result indicates that a spreadsheet interface, which has been essentially unchanged in decades, might provide a familiar and easy to use medium to view and edit tabular data. We found no evidence of vandalism in the maintenance strategies after analyzing unpaid contributor's edits, their number of interactions, and visits. A possible reason for this is the maintenance strategies, and our tabular dataset is of interest to the community, thus providing community-based motivation to improve its accuracy.

5.4 Takeaways: Maintenance Strategies

Maintenance strategies relying on unpaid contributors produce consistently accurate edits. Our study's longitudinal nature demonstrates the potential to maintain existing in-flux tabular datasets over long periods compared to prior work [1, 77] relying on unpaid contributors over short periods.

Unpaid contributors provided accurate edits per data type (column), regardless of whether they required domain-specific knowledge (i.e., Rank and Subfield), and thus builds upon past research of paid crowdworkers [23, 71]. This finding shows the potential to rely on user's altruistic motivations and mutual interests instead of paying crowdworkers to correct tabular data inaccuracies.

Unpaid contributors were highly accurate and active at filling in empty cells. Their prior knowledge could construe them as "experts", helping to explain why their edits increased accuracy [45]. This observation repeated itself whether the system solicited an unpaid contributor to edit data or not.

From our long-term observations as dataset curators running maintenance strategies, the unsolicited, unpaid contributors did not contribute new rows (i.e., professors) at a high enough rate to account for the number of new professors hired over three years. Similar research, either did not allow users to add new rows or the study was short lived and did not allow for the data to naturally evolve over time to study if users will contribute enough new rows [59, 77]. Systems that run long-term maintenance strategies should reduce the effort required to add new rows of data through interface design and limit the minimum number of required fields to add a new row.

Soliciting unpaid contributors to review and correct data relevant to their interests increased edits' overall accuracy. This finding confirms prior short-term research [77] and provides a long-term mechanism to increase a tabular dataset's longevity. A possible reason soliciting based on unpaid contributors' interests is beneficial is that they solicited while completing natural information-seeking goals, such as finding a possible advisor [31].

Contrary to prior work on Knowledge Graphs and Wikidata [25, 42], our maintenance strategies, focusing on collaborative efforts editing tabular data in spreadsheets, show that the number of visits per unpaid contributor was not predictive of vandalism or inaccurate edits. By integrating

tools to automatically translate and import tabular data to Wikidata [54, 60, 73], our work could generalize from focusing strictly on a spreadsheet interface to serving as a potential method to maintain tabular data from Wikipedia and Wikidata.

6 DISCUSSION

This work examines two case studies on verification and maintenance strategies for improving the accuracy of in-flux tabular datasets. Verification strategies require money to pay crowdworkers. We control for the time to wait for edits in verification strategies by limiting them to a short 2-week period. In contrast, maintenance strategies require time to wait for an unpaid contributor to visit the dataset and make edits. We control for money with these maintenance strategies, as hosting the web application to edit the dataset is the only cost. Our initial findings motivate and inform future work that can directly compare paid micro-task work strategies and those that rely on unpaid efforts. We will discuss several future ideas in the context of the findings we have presented and prior work.

6.1 Selecting the Best Approach: Paid or Unpaid

Choosing whether to use a pure paid crowdsourcing approach like our verification strategies or to adopt a peer production system like our maintenance strategies is difficult. Verification and maintenance strategies are not immune to incurring monetary costs or additional time for the dataset's curators. While running maintenance incurs minimal server costs, in verification, the time required to wait for a crowdworker to accept a paid micro-task can depend on the payment [38]. For example, requesters in verification have to approve completed tasks, post new ones, and respond to paid crowdworkers. This might not be ideal during a continuous data maintenance effort lasting years. Nevertheless, this amount of time is significantly less than the time required to curate a dataset using maintenance strategies. Maintenance strategies require time to passively wait for edits or employ other methods to attract visitors. A dataset curator will need to know how much their time is worth and choose the best approach for their needs. If a dataset does not contain data that evolves, it might be faster to use paid crowdworkers to verify existing data rather than wait. A better approach might be to mix intrinsic and extrinsic motivators in a single system.

Prior work has shown how mixing extrinsic and intrinsic rewards can improve the outcomes of specific tasks [8, 20]. Our results reflect this prior finding, where paid or unpaid users can generate better outcomes for specific tasks. For example, intrinsically motivated users were more effective at open-ended tasks. For example, unpaid contributors were more accurate in editing data that is more subjective or difficult to find, like Subfield or Join Year.

Another alternative approach to hosting a system to maintain a tabular dataset is to use one of the many tools to automatically translate and submit tabular data to Wikidata [54, 73]. Future work could assess if this unpaid approach leveraging popular platforms and tools, such as Wikidata, is more effective than maintaining a tabular dataset on a smaller, homegrown platform. Our maintenance strategies are one example of using a lesser-known platform to host a tabular dataset focused on one topic. Future research could assess whether a tabular dataset's maturity phase would benefit more from a popular platform than a smaller homegrown system. From our experience developing a system to maintain tabular data, its most important advantage is the ability to customize and adapt features to a specific dataset quickly.

Information can be subjective and difficult to interpret and label [10, 16]. This can be especially difficult in tabular datasets containing data types requiring expert knowledge with no accompanying information like Wikipedia to help users quickly understand context [77]. Requesters in verification strategies had to evaluate paid crowdworkers editing existing data to re-recruit them for knowledge-specific tasks. In contrast, unpaid contributor's prior knowledge and interest benefited them when

editing these data types. The verification strategies could benefit from deploying fast, low-cost tasks to pre-filter possible workers. A more ambitious idea would be for crowdsourcing platforms, such as Amazon Mechanical Turk, to allow requesters to find crowdworkers interested in the content of the tabular dataset. This would be analogous to our maintenance strategies benefiting from the user's interest in the data. By appealing to crowdworkers' intrinsic motivations and interests, they might be more motivated to find and interpret complex data types. This design implication also does not require requesters to develop more complex gamification mechanisms to improve results [44].

6.2 Hybrid Models to Compare Paid Crowdworkers and Unpaid Contributors

Our findings highlight future studies' potential to directly compare the collective efforts of paid crowdworkers and unpaid contributors' in controlled settings.

One method to exert more control is introducing errors to a tabular dataset intentionally and simultaneously recruiting paid crowdworkers and unpaid contributors and comparing their interactions, accuracy, and time required to correct errors. This would enable several valuable observations. For example, crowdworkers might take fewer actions to submit an edit since they want to progress quickly. In contrast, unpaid contributors might be exploring the dataset and only edit the information if they notice an error. Future work could expand the user interest profile used in the maintenance strategies with new types of interactions from paid crowdworkers and unpaid contributors. For example, the system could extract user interest from gaze information [27]. Running eye-tracking studies at scale in-the-wild is a recent advancement [58] to enable this type of research. With each piece of information carefully organized in cells, tabular data is an exciting interface to extract gaze information. Building on this idea, could each group's interactions before editing help determine their accuracy automatically? Automatically rating edits' accuracy to tabular data using interactions is an ongoing research question [77].

Could paid crowdworkers be motivated through targeted non-paid requests while editing to contribute to the tabular dataset? This is plausible if they believe these contributions are for the greater good. For example, Rogstadius et al. [67] studied how intrinsic motivators, such as completing tasks that contribute to a knowledge base, positively influence work quality compared to extrinsically-motivated tasks, such as monetary incentives. On the contrary, could unpaid contributors be motivated through a pay-per-task model? Previous researchers introduced an extrinsic motivator as an intervention for users who were initially intrinsically motivated to contribute [33, 78]. They found that this intervention did not impact the quality of work. Thus, the idea of using money to extrinsically motivate users of an open online system might not be beneficial for the maturity phase of the information within the system.

While our verification strategies rely on money to extrinsically motivate paid crowdworkers, our maintenance strategies more so resemble "micro-volunteering." Bernstein et al. define micro-volunteering, an example of altruistic motivation, as the process of completing small online tasks for social good [4]. Our unpaid contributors possibly visit and contribute edits because of an interest in Computer Science. While we did not explicitly advertise the system to our friends, they possibly have visited the dataset. If we recruit our friends, this would resemble "friendsourcing," where volunteers are recruited from a network of friends to make voluntary contributions [7]. Another possible source of unpaid contributors is professors themselves. We have received numerous requests for copies of the data so others can analyze hiring trends. Thus, it is reasonable to assume that some professors might be editing their information. We view this as a potential benefit of maintaining a dataset of public information that individuals might be personally interested in. Future work could compare unpaid contributions from different recruitment strategies, like micro-volunteering or friendsourcing. It would be reasonable to assume that friends share similar interests and might

freely contribute to datasets that are relevant to their friends. However, it is unclear whether friendsourcing is adequate for maintaining a tabular dataset's accuracy in the long run.

Another approach to tease apart the motivations for unpaid and paid contributions is randomly asking users why they contributed after an edit. For example, maybe they are correcting their data, or they were paid to correct data. This in-the-moment feedback method has proven successful in prior research [9, 76] and could be helpful to understand user's intrinsic and extrinsic motivators. Also, these motivators might differ across tabular datasets covering different topics.

These future directions build on our findings and prior literature to create hybrid strategies to maintain accurate datasets. The ability to continuously and cost-effectively recruit the right editors could enable tabular datasets to be perpetually maintained.

6.3 Automated Methods to Attract Unpaid Contributors

One of our goals is to learn different approaches for creating self-sustaining in-flux tabular datasets. The maintenance strategies do not need consistent effort to recruit and manage crowdworkers. However, they do require consistent numbers of interested visitors to contribute edits. Paying crowdworkers can generate edits quickly, but our work's initial observations show that this might be less effective than relying on unpaid contributors. We found it challenging to initiate a "start now" process for maintenance because the dataset's discovery is often serendipitous.

Wikipedia represents a success story in targeting and perpetually attracting unpaid contributions [64]. With free-form datasets such as articles, users can quickly and effortlessly discover facts from online sources of plain text, tables, or figures. In that process, they encounter incorrect or missing information. In contrast, users must perform multiple sorting and filtering operations with lengthy tabular datasets to find information. In this workflow, their focus rarely veers from their original query to new potentially interesting information, in need of edits, or missing information.

Large tabular datasets similar to those highlighted in our case studies that feature hundreds to thousands of rows of data and multiple columns have a potential advantage compared to unstructured articles or small tables on Wikipedia. This advantage relies on analyzing larger amount of data on a single cohesive topic, compared to analyzing smaller tables present on Wikipedia [39]. A platform hosting a single large tabular dataset can become more attractive to visitors by automatically generating and sharing insights and facts extracted from large amounts of data. This creation and sharing of knowledge could make a large tabular dataset more attractive, and potentially increase unpaid contributions [49].

Future work could create a system to generate sentences that describe insights and facts automatically based on statistical measures like maximums, modes, means, and outliers. This may also contain quantitative comparisons between the same values in columns or time-series data. These statistics could be automatically inputted into simple sentence templates, creating a fact to attract users. SuggestBot shows how to structure these fact-based sentences to maximize their effectiveness [12]. Structuring sentences in a "High-Involvement style" has also been shown to increase long-term retention among paid crowdworkers engaging in conversational micro-tasks [61]. By building on these ideas, one could structure these facts to attract, engage, and retain users. These human-consumable insights derived from large tabular datasets would move beyond prior research which focuses on finding tables related to other tables in Wikipedia [19].

A tabular dataset could market itself by automatically sharing these facts by posting them to relevant conversations in social media to attract regular visitors. In a similar effort, Botivist uses Twitter bots to recruit users to take action regarding Latin America's corruption issues [70]. This idea draws interesting parallels for recruiting unpaid contributors for maintenance. In our case, the system could share facts related to academia, Computer Science, or graduate schools to attract and motivate new users. By studying user interactions, one could determine which automatic

methods of generating "interesting" facts attracts the most substantial traffic and result in quality engagement, exemplified in views, and accurate edits of data related to the insight. If successful, this would allow the maintenance strategies to attract unpaid contributors to perpetually tabular datasets automatically.

7 CONCLUSION

Our work demonstrates how selecting the right verification strategy for different data types can yield value in money spent to acquire accurate edits. In particular, *Expert Rule* proved cost-effective for requesters, providing accurate edits at a lower cost than other paid verification strategies. It allows a dataset curator to improve the accuracy of a tabular dataset in a short period if the pay-per-task is high enough to attract and recruit trusted crowdworkers.

Continuous maintenance strategies enable groups of unpaid visitors to contribute edits freely or ask unpaid visitors to correct data that match their interests based on relevance. These approaches produced consistently accurate edits across all data types, indicating the potential of applying these approaches for self-sustaining tabular datasets. These continuous maintenance approaches can benefit tabular datasets where a row of data is viable for long periods, but individual cells may go out of date.

Our work covers one type of tabular data in the form of academic profiles. While this in-flux tabular dataset contains different levels of complexity and subjectivity, future work is necessary to explore even more complex tabular datasets. Over our various pilot studies for this work and related efforts, we have witnessed how the size (i.e., number of rows or columns) and complexity of data types can influence potential contributions.

Overall, this work lays a foundation for different strategies to mature accurate in-flux tabular datasets. Accurate tabular datasets can yield exciting insights across fields, and ensuring their accuracy and longevity is essential to providing continuous insights to society. This work can help future researchers develop hybrid approaches to efficiently and cost-effectively collect, verify, and maintain their tabular datasets.

8 ACKNOWLEDGMENTS

These case studies would not be possible without numerous contributions from many people across six years. First, we want to thank the Brown University students who crowdsourced the data in the HCI seminar in Spring 2014, Spring 2015, and Spring 2018. Lucas Kang for reviewing the data in the summer of 2014 and Brendan Le for reviewing earlier versions of this paper. Also, Lucy Van Kleunan, Marianne Aubin-Le Quere, Abraham Peterkin, and Yirui Huang for helping to shape the system used to run the Maintenance Strategies. Lastly, the numerous online visitors and paid crowdworkers who reviewed and corrected the data.

REFERENCES

[1] Bryan Alexander. 2020. The little spreadsheet that could, and did: crowdsourcing COVID-19, higher education, data, and stories. https://bryanalexander.org/research-topics/the-little-spreadsheet-that-could-and-did-crowdsourcing-covid-19-higher-education-data-and-stories/. (Accessed on 04/12/2021).

[2] Roland Benabou and Jean Tirole. 2003. Intrinsic and extrinsic motivation. *The review of economic studies* 70, 3 (2003), 489–520.

[3] Emery D. Berger. 2020. *CSRankings.* Retrieved April 27, 2020 from http://csrankings.org/

[4] Michael Bernstein, Mike Bright, Ed Cutrell, Steven Dow, Elizabeth Gerber, Anupam Jain, and Anand Kulkarni. 2013. Micro-volunteering: helping the helpers in development. In *Proceedings of the 2013 conference on Computer supported cooperative work companion.* ACM, New York, NY, USA, 85–88.

[5] Michael S Bernstein, Greg Little, Robert C Miller, Björn Hartmann, Mark S Ackerman, David R Karger, David Crowell, and Katrina Panovich. 2010. Soylent: a word processor with a crowd inside. In *Proceedings of the 23nd annual ACM symposium on User interface software and technology.* ACM, New York, NY, USA, 313–322.

[6] Leon Bornemann, Tobias Bleifuß, Dmitri V Kalashnikov, Felix Naumann, and Divesh Srivastava. 2020. Natural key discovery in Wikipedia tables. In *Proceedings of The Web Conference 2020*. ACM, New York, NY, USA, 2789–2795.

[7] Erin Brady, Meredith Ringel Morris, and Jeffrey P Bigham. 2015. Gauging receptiveness to social microvolunteering. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. ACM, New York, NY, USA, 1055–1064.

[8] Gordon Burtch, Yili Hong, Ravi Bapna, and Vladas Griskevicius. 2018. Stimulating online reviews by combining financial incentives and social norms. *Management Science* 64, 5 (2018), 2065–2082.

[9] Ana Caraban, Evangelos Karapanos, Daniel Gonçalves, and Pedro Campos. 2019. 23 ways to nudge: A review of technology-mediated nudging in human-computer interaction. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, USA, 1–15.

[10] John Joon Young Chung, Jean Y Song, Sindhu Kutty, Sungsoo Hong, Juho Kim, and Walter S Lasecki. 2019. Efficient elicitation approaches to estimate collective crowd answers. *Proceedings of the ACM on Human-Computer Interaction* 3, CSCW (2019), 1–25.

[11] Gregorio Convertino and Andy Echenique. 2017. Self-service data preparation and analysis by business users: New needs, skills, and tools. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*. ACM, New York, NY, USA, 1075–1083.

[12] Dan Cosley, Dan Frankowski, Loren Terveen, and John Riedl. 2007. SuggestBot: Using Intelligent Task Routing to Help People Find Work in Wikipedia. In *Proceedings of the 12th International Conference on Intelligent User Interfaces* (Honolulu, Hawaii, USA) *(IUI '07)*. ACM, New York, NY, USA, 32–41. https://doi.org/10.1145/1216295.1216309

[13] Florian Daniel, Pavel Kucherbaev, Cinzia Cappiello, Boualem Benatallah, and Mohammad Allahbakhsh. 2018. Quality control in crowdsourcing: A survey of quality attributes, assessment techniques, and assurance actions. *ACM Computing Surveys (CSUR)* 51, 1 (2018), 1–40.

[14] Edward L Deci and Richard M Ryan. 1991. A motivational approach to self: Integration in personality. *In Nebraska Symposium on Motivation: Perspectives on Motivation* 38 (1991), 237–288.

[15] Anhai Doan, Raghu Ramakrishnan, and Alon Y. Halevy. 2011. Crowdsourcing Systems on the World-Wide Web. *Commun. ACM* 54, 4 (April 2011), 86–96. https://doi.org/10.1145/1924421.1924442

[16] Anca Dumitrache. 2015. Crowdsourcing disagreement for collecting semantic annotation. In *Proc. ESWC*. Springer, New York, NY, USA, 701–710.

[17] Serge Egelman, Ed H. Chi, and Steven Dow. 2014. *Ways of Knowing in HCI.* Springer New York, New York, NY, Chapter Crowdsourcing in HCI Research, 267–289.

[18] Siamak Faridani, Björn Hartmann, and Panagiotis G. Ipeirotis. 2011. What's the Right Price? Pricing Tasks for Finishing on Time. In *Proceedings of the 11th AAAI Conference on Human Computation (AAAIWS'11-11)*. AAAI, Menlo Park, CA, USA, 26–31. http://dl.acm.org/citation.cfm?id=2908698.2908703

[19] Besnik Fetahu, Avishek Anand, and Maria Koutraki. 2019. Tablenet: An approach for determining fine-grained relations for wikipedia tables. In *The World Wide Web Conference*. ACM, New York, NY, USA, 2736–2742.

[20] Claudia Flores-Saviaga, Ricardo Granados, Liliana Savage, Lizbeth Escobedo, and Saiph Savage. 2020. Understanding the complementary nature of paid and volunteer crowds for content creation. *Avances en Interacción Humano-Computadora* 1, 1 (2020), 37–44.

[21] Mary Jo Foley. 2010. About that 1 billion Microsoft Office figure... https://www.zdnet.com/article/about-that-1-billion-microsoft-office-figure/. [Online; accessed 2020-04-20].

[22] Ujwal Gadiraju, Ricardo Kawase, Stefan Dietze, and Gianluca Demartini. 2015. Understanding Malicious Behavior in Crowdsourcing Platforms: The Case of Online Surveys. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (Seoul, Republic of Korea) *(CHI '15)*. Association for Computing Machinery, New York, NY, USA, 1631–1640. https://doi.org/10.1145/2702123.2702443

[23] Daniel Haas, Jason Ansel, Lydia Gu, and Adam Marcus. 2015. Argonaut: Macrotask Crowdsourcing for Complex Data Processing. *Proceedings of the VLDB Endowment* 8, 12 (Aug. 2015), 1642–1653. https://doi.org/10.14778/2824032.2824062

[24] Daniel Haas, Sanjay Krishnan, Jiannan Wang, Michael J. Franklin, and Eugene Wu. 2015. Wisteria: Nurturing Scalable Data Cleaning Infrastructure. *Proceedings of the VLDB Endowment* 8, 12 (Aug. 2015), 2004–2007. https://doi.org/10.14778/2824032.2824122

[25] Stefan Heindorf, Martin Potthast, Benno Stein, and Gregor Engels. 2016. Vandalism detection in wikidata. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*. ACM, New York, NY, USA, 327–336.

[26] Thomas Herndon, Michael Ash, and Robert Pollin. 2014. Does high public debt consistently stifle economic growth? A critique of Reinhart and Rogoff. *Cambridge journal of economics* 38, 2 (2014), 257–279.

[27] Daniel Hienert, Dagmar Kern, Matthew Mitsui, Chirag Shah, and Nicholas J Belkin. 2019. Reading protocol: Understanding what has been read in interactive information retrieval tasks. In *Proceedings of the 2019 Conference on Human Information Interaction and Retrieval (CHIIR '19)*. ACM, New York, NY, USA, 73–81.

[28] Benjamin Mako Hill and Aaron Shaw. 2020. Wikipedia and the End of Open Collaboration. *Wikipedia* 20 (2020).
[29] Matthias Hirth, Tobias Hoßfeld, and Phuoc Tran-Gia. 2013. Analyzing costs and accuracy of validation mechanisms for crowdsourcing platforms. *Mathematical and Computer Modelling* 57, 11-12 (2013), 2918–2932.
[30] Chien-Ju Ho, Aleksandrs Slivkins, Siddharth Suri, and Jennifer Wortman Vaughan. 2015. Incentivizing high quality crowdwork. In *Proceedings of the 24th International Conference on World Wide Web*. ACM, New York, NY, USA, 419–429.
[31] Orland Hoeber, Anoop Sarkar, Andrei Vacariu, Max Whitney, Manali Gaikwad, and Gursimran Kaur. 2017. Evaluating the value of Lensing Wikipedia during the information seeking process. In *Proceedings of the 2017 Conference on Conference Human Information Interaction and Retrieval*. ACM, New York, NY, USA, 77–86.
[32] Ting-Hao Kenneth Huang and Jeffrey P Bigham. 2017. A 10-Month-Long Deployment Study of On-Demand Recruiting for Low-Latency Crowdsourcing.. In *HCOMP*. AAAI, Menlo Park, CA, USA, 61–70.
[33] Warut Khern-am nuai, Karthik Kannan, and Hossein Ghasemkhani. 2018. Extrinsic versus intrinsic rewards for contributing reviews in an online platform. *Information Systems Research* 29, 4 (2018), 871–892.
[34] Johannes Kiesel, Martin Potthast, Matthias Hagen, and Benno Stein. 2017. Spatio-temporal analysis of reverted wikipedia edits. In *Proceedings of the International AAAI Conference on Web and Social Media*, Vol. 11. AAAI, Menlo Park, CA, USA, 122–131.
[35] Aniket Kittur, Ed H. Chi, and Bongwon Suh. 2008. Crowdsourcing User Studies with Mechanical Turk. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Florence, Italy) *(CHI '08)*. ACM, New York, NY, USA, 453–456. https://doi.org/10.1145/1357054.1357127
[36] Aniket Kittur, Jeffrey V Nickerson, Michael Bernstein, Elizabeth Gerber, Aaron Shaw, John Zimmerman, Matt Lease, and John Horton. 2013. The future of crowd work. In *Proceedings of the 2013 Conference on Computer Supported Cooperative Work*. ACM, New York, NY, USA, 1301–1318.
[37] Aniket Kittur, Boris Smus, Susheel Khamkar, and Robert E. Kraut. 2011. CrowdForge: Crowdsourcing Complex Work. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology* (Santa Barbara, California, USA) *(UIST '11)*. ACM, New York, NY, USA, 43–52. https://doi.org/10.1145/2047196.2047202
[38] Thomas Kohler. 2015. Crowdsourcing-based business models: how to create and capture value. *California Management Review* 57, 4 (2015), 63–84.
[39] Flip Korn, Xuezhi Wang, You Wu, and Cong Yu. 2019. Automatically generating interesting facts from wikipedia tables. In *Proceedings of the 2019 International Conference on Management of Data*. ACM, New York, NY, USA, 349–361.
[40] Michel Krieger, Emily Margarete Stark, and Scott R. Klemmer. 2009. Coordinating Tasks on the Commons: Designing for Personal Goals, Expertise and Serendipity. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Boston, MA, USA) *(CHI '09)*. ACM, New York, NY, USA, 1485–1494. https://doi.org/10.1145/1518701.1518927
[41] Anand Kulkarni, Matthew Can, and Björn Hartmann. 2012. Collaboratively Crowdsourcing Workflows with Turkomatic. In *Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work* (Seattle, Washington, USA) *(CSCW '12)*. ACM, New York, NY, USA, 1003–1012. https://doi.org/10.1145/2145204.2145354
[42] Srijan Kumar, Francesca Spezzano, and VS Subrahmanian. 2015. Vews: A wikipedia vandal early warning system. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, New York, NY, USA, 607–616.
[43] Natalie Kupferberg and Bridget McCrate Protus. 2011. Accuracy and completeness of drug information in Wikipedia: an assessment. *Journal of the Medical Library Association: JMLA* 99, 4 (2011), 310.
[44] Sascha Lichtenberg, Tim-Benjamin Lembcke, Mattheus Brening, Alfred Benedikt Brendel, and Simon Trang. 2020. Can Gamification lead to Increase Paid Crowdworkers Output?. In *Wirtschaftsinformatik (Zentrale Tracks)*. 1188–1202.
[45] Roman Lukyanenko, Jeffrey Parsons, Yolanda F Wiersma, and Mahed Maddah. 2019. Expecting the unexpected: Effects of data collection design choices on the quality of crowdsourced user-generated content. *MIS Quarterly* 43, 2 (2019), 623–648.
[46] Andrew Mao, Ece Kamar, Yiling Chen, Eric Horvitz, Megan E Schwamb, Chris J Lintott, and Arfon M Smith. 2013. Volunteering versus work for pay: Incentives and tradeoffs in crowdsourcing. In *First AAAI conference on human computation and crowdsourcing*. AAAI, Menlo Park, CA, USA, 94–102.
[47] Adam Marcus. 2013. How I Learned to Stop Worrying and Love the Crowd. https://www.youtube.com/watch?v=FL9Q43zO1BQ (Online; accessed on 2020-04-20).
[48] Winter Mason and Duncan J. Watts. 2009. Financial Incentives and the "Performance of Crowds". In *Proceedings of the ACM SIGKDD Workshop on Human Computation* (Paris, France) *(HCOMP '09)*. ACM, New York, NY, USA, 77–85. https://doi.org/10.1145/1600150.1600175
[49] Paweł Mikołajczak and Piotr Bajak. 2021. Does NGOs' commercialization affect volunteer work? The crowding out or crowding in Effect. *Public Organization Review* 21, 1 (2021), 103–118.
[50] Joshua D Miller, Michael Crowe, Brandon Weiss, Jessica L Maples-Keller, and Donald R Lynam. 2017. Using online, crowdsourcing platforms for data collection in personality disorder research: The example of Amazon's Mechanical Turk. *Personality Disorders: Theory, Research, and Treatment* 8, 1 (2017), 26.

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

[52] Meredith Ringel Morris, Jaime Teevan, and Katrina Panovich. 2010. What Do People Ask Their Social Networks, and Why?: A Survey Study of Status Message Q&A Behavior. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Atlanta, Georgia, USA) *(CHI '10)*. ACM, New York, NY, USA, 1739–1748. https://doi.org/10.1145/1753326.1753587

[53] Babak Naderi. 2018. *Motivation of workers on microtask crowdsourcing platforms*. Springer, New York, NY, USA.

[54] Phuc Nguyen, Ikuya Yamada, Natthawut Kertkeidkachorn, Ryutaro Ichise, and Hideaki Takeda. 2020. MTab4Wikidata at SemTab 2020: Tabular Data Annotation with Wikidata.. In *SemTab@ ISWC*. ACM, New York, NY, USA, 86–95.

[55] Jasper Oosterman, Archana Nottamkandath, Chris Dijkshoorn, Alessandro Bozzon, Geert-Jan Houben, and Lora Aroyo. 2014. Crowdsourcing Knowledge-intensive Tasks in Cultural Heritage. In *Proceedings of the 2014 ACM Conference on Web Science* (Bloomington, Indiana, USA) *(WebSci '14)*. ACM, New York, NY, USA, 267–268. https://doi.org/10.1145/2615569.2615644

[56] Raymond R Panko. 1998. What we know about spreadsheet errors. *Journal of Organizational and End User Computing (JOEUC)* 10, 2 (1998), 15–21.

[57] Alexandra Papoutsaki, Hua Guo, Danae Metaxa-Kakavouli, Connor Gramazio, Jeff Rasley, Wenting Xie, Guan Wang, and Jeff Huang. 2015. Crowdsourcing from Scratch: A Pragmatic Experiment in Data Collection by Novice Requesters. In *Proceedings of the Third AAAI Conference on Human Computation and Crowdsourcing (HCOMP)*. AAAI, Menlo Park, CA, USA, 140–149.

[58] Alexandra Papoutsaki, James Laskey, and Jeff Huang. 2017. Searchgazer: Webcam eye tracking for remote studies of web search. In *Proceedings of the 2017 Conference on Conference Human Information Interaction and Retrieval (CHIIR '17)*. ACM, New York, NY, USA, 17–26.

[59] Hyunjung Park and Jennifer Widom. 2014. CrowdFill: Collecting Structured Data from the Crowd. In *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data* (Snowbird, Utah, USA) *(SIGMOD '14)*. ACM, New York, NY, USA, 577–588. https://doi.org/10.1145/2588555.2610503

[60] Wikimedia Project. 2021. Wikipedia and Wikidata Tools - Meta. https://meta.wikimedia.org/wiki/Wikipedia_and_Wikidata_Tools. (Accessed on 07/10/2021).

[61] Sihang Qiu, Ujwal Gadiraju, and Alessandro Bozzon. 2020. Improving worker engagement through conversational microtask crowdsourcing. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, USA, 1–12.

[62] Giovanni Quattrone, Martin Dittus, and Licia Capra. 2017. Work Always in Progress: Analysing Maintenance Practices in Spatial Crowd-sourced Datasets. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing* (Portland, Oregon, USA) *(CSCW '17)*. ACM, New York, NY, USA, 1876–1889. https://doi.org/10.1145/2998181.2998267

[63] Alexander J. Quinn and Benjamin B. Bederson. 2011. Human Computation: A Survey and Taxonomy of a Growing Field. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Vancouver, BC, Canada) *(CHI '11)*. Association for Computing Machinery, New York, NY, USA, 1403–1412. https://doi.org/10.1145/1978942.1979148

[64] Sam Ransbotham and Gerald C Kane. 2011. Membership turnover and collaboration success in online communities: Explaining rises and falls from grace in Wikipedia. *Mis Quarterly* 35 (2011), 613–627.

[65] Katharina Reinecke and Krzysztof Z Gajos. 2015. LabintheWild: Conducting large-scale online experiments with uncompensated samples. In *Proceedings of the 18th ACM conference on computer supported cooperative work & social computing*. ACM, New York, NY, USA, 1364–1378.

[66] Ganit Richter, Daphne R Raban, and Sheizaf Rafaeli. 2015. Studying gamification: the effect of rewards and incentives on motivation. In *Gamification in education and business*. Springer, New York, NY, USA, 21–46.

[67] Jakob Rogstadius, Vassilis Kostakos, Aniket Kittur, Boris Smus, Jim Laredo, and Maja Vukovic. 2011. An assessment of intrinsic and extrinsic motivation on task performance in crowdsourcing markets. In *Fifth International AAAI Conference on Weblogs and Social Media*. AAAI, Menlo Park, CA, USA, 321–328.

[68] Richard M Ryan and Edward L Deci. 2000. Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being. *American psychologist* 55, 1 (2000), 68.

[69] Elliot Salisbury, Ece Kamar, and Meredith Ringel Morris. 2017. Toward scalable social alt text: Conversational crowdsourcing as a tool for refining vision-to-language technology for the blind. In *Fifth AAAI Conference on Human Computation and Crowdsourcing (HCOMP)*. AAAI, Menlo Park, CA, USA, 147–156.

[70] Saiph Savage, Andres Monroy-Hernandez, and Tobias Höllerer. 2016. Botivist: Calling volunteers to action using online bots. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing*. ACM, New York, NY, USA, 813–822.

[71] Linda See, Alexis Comber, Carl Salk, Steffen Fritz, Marijn van der Velde, Christoph Perger, Christian Schill, Ian McCallum, Florian Kraxner, and Michael Obersteiner. 2013. Comparing the quality of crowdsourced data contributed by expert and non-experts. *PloS one* 8, 7 (2013), e69958.
[72] C Estelle Smith, Bowen Yu, Anjali Srivastava, Aaron Halfaker, Loren Terveen, and Haiyi Zhu. 2020. Keeping Community in the Loop: Understanding Wikipedia Stakeholder Values for Machine Learning-Based Systems. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. ACM, New York, NY, USA, 1–14.
[73] Pedro A Szekely, Daniel Garijo, Jay Pujara, Divij Bhatia, and Jiasheng Wu. 2019. T2WML: A Cell-Based Language to Map Tables into Wikidata Records.. In *ISWC Satellites*. ACM, New York, NY, USA, 45–48.
[74] Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledge base. *Commun. ACM* 57 (2014), 78–85.
[75] W3C. 2015. Model for Tabular Data and Metadata on the Web. https://www.w3.org/TR/tabular-data-model/. [Online; accessed 2020-04-20].
[76] Shaun Wallace, Brendan Le, Luis A Leiva, Aman Haq, Ari Kintisch, Gabrielle Bufrem, Linda Chang, and Jeff Huang. 2020. Sketchy: Drawing inspiration from the crowd. *Proceedings of the ACM on Human-Computer Interaction* 4, CSCW2 (2020), 1–27.
[77] Shaun Wallace, Lucy Van Kleunen, Marianne Aubin-Le Quere, Abraham Peterkin, Yirui Huang, and Jeff Huang. 2017. Drafty: Enlisting Users to be Editors who Maintain Structured Data. In *Proceedings of the Fifth AAAI Conference on Human Computation and Crowdsourcing (HCOMP)*. AAAI, Menlo Park, CA, USA, 187–196.
[78] Jing Wang, Gen Li, and Kai-Lung Hui. 2018. Do Monetary Incentives Create a Spillover Effect on Free Knowledge Contribution? Evidence from a Natural Experiment. *Evidence from a Natural Experiment (June 25, 2018)* (2018), 1–24.
[79] Andrea Wiggins, Greg Newman, Robert D Stevenson, and Kevin Crowston. 2011. Mechanisms for data quality and validation in citizen science. In *e-Science Workshops (eScienceW), 2011 IEEE Seventh International Conference on*. IEEE, Piscataway, NJ, USA, 14–19.
[80] Dennis M Wilkinson and Bernardo A Huberman. 2007. Assessing the value of coooperation in wikipedia. *arXiv preprint cs/0702140* (2007), 1–14.
[81] Ying Xu and Carleen Maitland. 2019. Participatory Data Collection and Management in Low-resource Contexts: A Field Trial with Urban Refugees. In *Proceedings of the Tenth International Conference on Information and Communication Technologies and Development* (Ahmedabad, India) *(ICTD '19)*. ACM, New York, NY, USA, Article 18, 12 pages. https://doi.org/10.1145/3287098.3287104
[82] Omar F Zaidan and Chris Callison-Burch. 2011. Crowdsourcing translation: Professional quality from non-professionals. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*. Association for Computational Linguistics, Stroudsburg, PA, USA, 1220–1229.
[83] Shuo Zhang and Krisztian Balog. 2020. Web Table Extraction, Retrieval, and Augmentation: A Survey. *ACM Transactions on Intelligent Systems and Technology (TIST)* 11, 2 (2020), 1–35.
[84] Zhe Zhao, Zhiyuan Cheng, Lichan Hong, and Ed H Chi. 2015. Improving user topic interest profiles by behavior factorization. In *Proceedings of the 24th International Conference on World Wide Web*. ACM, New York, NY, USA, 1406–1416.

Received April 2021; accepted July 2021