CrowdAnalytics: Support for Redundancy Clustering and Provenance in Crowdsourced Data Analysis

Shiry Ginosar  Avital Steinitz

Abstract—Crowdsourcing can help analysts make sense of large datasets by using paid workers to perform analysis tasks like examining visualizations, generating explanations, and finding sources that support them. However, explanations generated by multiple workers are often redundant and the quality of explanations can vary heavily depending on the workers who generated them and the sources they used. To utilize crowd work, analysts must be able to quickly make confidence judgments about the explanations that workers produce. We demonstrate two sets of techniques to help analysts manage crowdsourced explanations: (1) We use workers web searches and browsing histories, as well as explicit highlighting and source-review tasks, to provide analysts with detailed information about where and how each explanation was sourced. (2) We present crowdsourced and automated techniques to help analysts cluster redundant explanations and identify explanations that occur frequently. To support this process, we use a proxied web browser embedded inside crowdsourced explanation tasks. We expose this information in an explanation-management interface that allows analysts to interactively explore clustered explanations.

Index Terms—Crowdsourcing, Social Data Analysis

1 INTRODUCTION
As analysts explore and examine large datasets they often engage in an iterative process of sensemaking involving both visualizations and outside resources [7]. Generating explanations for trends and outliers in datasets and finding evidence to support them are key parts of this sensemaking loop and can entail considerable work for the analyst (examining charts, ideating, finding and confirming sources, etc.). Recent work has shown that using paid crowd workers can provide a scalable way of generating explanations and finding resources [13]. However, sets of explanations generated in parallel may contain redundant information and the quality of responses from crowd workers can be highly variable. Moreover, examining and verifying results generated by crowd workers can add additional work for an analyst.

Using explanations generated by workers can also be complicated by lack of provenance - information about how explanations were generated and what sources they rely on. For example, it may be unclear if a worker’s explanation derived from a reputable source or is the worker’s own speculation. In order to effectively leverage the crowd’s work, analysts need to quickly make confidence judgments about the explanations and citations that workers produce and decide which ones to pursue.

This paper focuses on helping analysts make sense of sets of explanations generated by crowd workers. We explore a range of signals that analysts may use to filter, organize, and make confidence judgments about explanations. We then demonstrate techniques to help analysts manage results from the crowd:

(1) We use workers’ web searches and browsing histories, as well as explicit highlighting and source-review tasks, to provide provenance for explanations. Analysts can use this information to make confidence judgments about explanations and decide when and where to explore further.

(2) We demonstrate crowdsourced and automated techniques to help analysts cluster redundant explanations. We propose and evaluate automated and crowd-based approaches for clustering explanations. Our results show that asking workers to cluster explanations directly produces the best clusters.

To this end, we use a proxied web browser embedded within crowdsourced tasks to monitor how workers complete tasks and track the sources that inform their explanations. We expose all of this information via an explanation-management interface that allows analysts to interactively explore clustered explanations and exposes provenance information.

2 RELATED WORK
Our work builds on prior work both in social data analysis and crowdsourcing.

Over the past decade, a number of social data analysis systems, including Sense.us [5], Many Eyes [11], and Pathfinder[6] have allowed large numbers of users to share, explore, and discuss data visualizations. These research systems, (as well as commercial social data analysis systems like ata360.org and the now-defunct Swivel.com) were premised on the idea that many users operating together can parallelize the process of analyzing datasets. While in practice, motivated users of these tools do explore and discuss extensively [12], most visualizations receive little attention and it is difficult for analysts to leverage other users’ effort in a consistent or systematic way.

However, evaluations of systems like CommentSpace [14] have shown that dividing visual analysis tasks into smaller pieces and explicitly dividing the analysis among participants can produce good analytic results. We extend this approach, breaking visual analysis into even smaller analysis microtasks and allowing analysts to systematically assign them to crowd workers. These small, granular pieces of work can be incorporated inside analysts’ existing workflows. This approach is inspired by systems like Soylent [3], which embeds crowd labor into existing word processing workflows.

Other recent work, has explored the application of distributed collaborators to sensemaking and analysis tasks. Fisher, et al. [4] examine how distributed collaborators iteratively build an understanding of information by organizing their work and the work of others. Researchers have also explored “instrumenting the crowd” [8] by augmenting crowd-based tasks to track workers’ behavior and automatically assess the quality of their work. We log worker activity, but use it to help analysts make assessments about explanation provenance.

3 USEFUL ASPECTS OF EXPLANATIONS
When considering explanations for a salient event, an analyst’s key task is to determine whether or not each explanation should be discarded, retained, or explored further. To do this, the analyst must determine whether she has confidence in an explanation. An analyst may consider several different aspects of an explanation to make this assessment:

Is the explanation text clear, specific, and plausible? Some fraction of workers in crowd marketplaces typically satisfies - performing the bare minimum amount of work to complete the task - and may generate poorly-constructed, unspecific, or logically implausible results. By comparison, well-written explanations that appear internally consistent can instill greater confidence.

Does the explanation cite a reputable source? If an explanation draws from a source the analyst is familiar with, the analyst can also
leverage her knowledge of the source to make confidence judgements about the explanation. Citing a source that an analyst trusts (for example a known news organization or reference) may bolster the explanation’s credibility, while an unknown or disreputable source may diminish it.

How frequently was an explanation given? In many cases, multiple workers produce variants of the same explanation. The analyst would wish to consider only a single version rather than reading all of the redundant responses. Additionally, if an explanation is proposed multiple times by different workers, it may indicate that the explanation is more likely a good one. Conversely, a lack of redundant explanations may signal that the cardinality of the space of likely answers is large and the odds that we’ve found the most likely answer is lower [10].

Does the content of the explanation come from the source or the worker? When assessing the provenance of an explanation it can be important to understand whether the facts and inferences included in the explanation were produced directly by the worker or were pulled from sources on the web. In our experience, workers who are not domain experts (including most workers on Mechanical Turk) are more adept at pulling good explanations from sources than producing them independently. As a result, explanations that repeat or paraphrase content from a source tend to be more credible than explanations based on facts produced entirely by the worker. As such, providing an indication of whether or not the content is copied or paraphrased directly from the source can aid analysts’ confidence judgements.

Is the explanation corroborated by multiple sources? Multiple versions of an explanation that cite different sources may increase confidence even further, since sources can corroborate one another. On the other hand, many copies of the same text appearing in multiple untrusted sources such as blogs may not instill confidence in the authenticity of the information content of the text.

4 Supporting the Analyst in His Task

We consider a workflow (Figure 1) for crowdsourcing social data analysis similar to [13]. In this workflow, an analyst manually or automatically selects views of a dataset that contain trends or other features of interest. The analyst then submits these charts to workers in a paid crowd marketplace who examine and evaluate them in parallel. These workers independently complete analysis microtasks in which they generate candidate explanations for why a trend or an outlier may have occurred, and provide links to web pages that support their explanation. Each microtask (Figure 2) displays a single chart along with a series of prompts.

Using this workflow, we can produce large numbers of high-quality explanations for a range of datasets. However, once the explanations are produced it is up to the analyst to examine each one and decide whether or not she is confident in its and wishes to keep or build on it. This can be time-consuming and may entail considerable effort.

Fortunately, analysts can leverage the fact that explanations are often redundant and are frequently supported by known sources on the web to prioritize or triage sets of redundant answers and make confidence assessments more quickly. Here we describe crowdsourced and automated methods for extracting provenance and redundancy information.

4.1 Example Deployment

In subsequent sections, we consider several approaches for assessing and clustering crowdsourced explanations. To illustrate and test these approaches, we conducted a deployment of our system using workers from Amazon Mechanical Turk (www.mturk.com), which we will refer to throughout our discussion. In this deployment, we asked workers to generate explanations for 12 charts drawn from 3 datasets covering a range of public-interest data types (US employment data for major metropolitan areas, graduation and earning statistics for major universities, and UN food price indices). We showed each chart to 10 different workers, for a total of 120 analysis microtasks. 93 workers participated, producing a corpus of 156 explanations.

4.2 Collecting Provenance Information

In order to make judgements about source reputability, paraphrasing, and corroboration, analysts need more detailed information about the websites workers visit and the specific source text they use to produce their results. To collect this information, we instrument the analysis microtasks that workers perform so that they provide a record of workers’ browsing activity. We also inject custom code into websites that workers visit as part of the task so that workers can highlight specific portions of source pages that support their explanations (Figure 2).

4.2.1 Logging Worker Activity

Recording the sites workers visit as they perform microtasks is difficult because the same-origin policy [1] implemented by modern web browsers prevents code from one internet domain from accessing web pages loaded from other domains. As a result, our microtasks cannot monitor activity that occurs in browser windows or tabs that do not originate from our site. We circumvent this restriction by having workers browse and search for sources using a custom web browser embedded within the analysis microtask (Figure 2B). This custom browser consists of a set of browser controls and an IFrame that loads web pages via our own custom proxy server. Serving and requesting pages via our own server (Figure 3) allows us to log each page a worker visits and monitor any web searches they make as they forage for sources and candidate explanations.

This proxy infrastructure has a few limitations. For technical and security reasons, we cannot proxy content served using protocols other than HTTP and cannot handle cookies. As a result, we cannot load content from sites that require users to authenticate or log in. Finally, we cannot guarantee that workers perform all of their browsing within our proxied interface and not in another browser window. However, our analysis of log data suggests that most of the sites workers visit are rendered appropriately via the proxy and that workers are active within our browser window for the majority of the time they spend on the task.

4.2.2 Supporting Fine-Grained Citations

Typically, when a worker cites a web page to support an explanation, only a small portion of the page (a paragraph or even a few sentences) is directly relevant to their explanation. Page-level citations can make it difficult for analysts or workers in rating microtasks to assess as source, since they may need to examine the entire web page to find the relevant text. Using the text of page-level citations to help cluster explanations can also be problematic since much of the page text may be unrelated to the explanation and individual pages often provide support for multiple different explanations. We support finer-grained source citations by allowing workers to mark specific blocks of text within pages as sources.

4.3 Detecting Redundancy

To help the analyst sift through the metadata created by the crowd we detect redundant explanations and cluster them together. Clustering
In a manual approach we explicitly ask human workers to group redundant explanations together. We contrast these three approaches by comparing their performances to a “ground truth” clustering we collected from three experts.

### 4.3.1 Automated Clustering

The first set of methods we discuss use a text-based measure of similarity between explanations as input to an automated clustering algorithm. Although a large number of different text similarity metrics have been developed for comparing and clustering pieces of text, the definition of similarity is often subjective and may vary depending on the task, context, and reader. Here we focus on three variations of syntactic metrics that emerge from mining textual data. In essence, we demonstrate how standard text mining algorithms can be used to detect redundancies in the explanations. To this end, we represent each explanation as a point in a high-dimensional feature space in one of the following ways.

Our basic representation constructs a Boolean Bag-of-Words from a textual explanation, as well as their associated URL and selected paragraph. Each explanation in this model is represented by a boolean vector, indexed by all the words in the lexicon. The representation of an explanation $e$ is a vector $x$, where the $w$th entry of $x$ equals one if and only if the word $w$ occurs at least once in the explanation. This representation is similar to the Vector Space Model but without taking into consideration the frequency in which words appear (see [9]). Since models based on word frequencies were designed for representing paragraphs, they seem unfitting to our short explanations that contain a couple of sentences at most, and often do not have any word repetitions in them.

Using this representation, we compute the similarity between a pair of explanations using Cosine Similarity which naturally compensates for the explanation length [9]. Since we are using a binomial model, this measure of similarity reduces to a measure of the number of words that co-occur in the two explanations. In order to prevent pairs of explanations from appearing similar based on the co-occurrence of uninformative words, we perform a manual pre-processing step and remove stop words and all words that appear in the outlier chart. Excluding repeated words, we perform a manual pre-processing step and remove stop words and all words that appear in the outlier chart. For example, if a city has a score of 5 during a month, it means that the number of people employed in that city was 15% larger during that month than in January 2000.

### 4.3.2 Semi-Automated Clustering

While automated text similarity methods are scalable and fast, they rely on the assumption that similar explanations will use similar language. These measures can fail when explanations use different terms to describe the same phenomenon or when the connection between two comments requires outside knowledge. Because detecting these sorts of relationships requires complex processing of natural language and outside context, human workers can make much better similarity assessments.

In our semi-automated clustering approach, we use crowd workers to evaluate pairs of explanations and score their similarity. We show workers two explanations at a time and ask them to decide if the two are redundant or not (Figure 4). Using multiple workers, we collect 5 judgments for every pair of explanations, then average the scores to produce an average similarity score for the pair. We then use these similarity judgments to cluster explanations using k-means.

While using worker judgments in lieu of word overlap provides a better measure of explanation similarity, this method also has several potential drawbacks. Assessing redundancy for all pairs requires 20 operations and the number of comparisons grows quickly as the number of explanations grows. Comparing only 10 explanations requires $\binom{10}{2} = 45$ comparisons and comparing 20 explanations requires $\binom{20}{2} = 190$. This number increases even further if scores from multiple workers (in our case, 5) are averaged to produce pairwise similarity scores.

### 4.3.3 Manual Clustering

Finally, we explore a manual clustering approach, in which individual workers examine complete sets of explanations rather than iso-
lated pairs and cluster them explicitly. This allows workers to evaluate groups of explanations together and identify larger themes.

To simplify the task of specifying clusters, we created a system where workers group comments by color-coding them. In each manual clustering task (Figure 5), workers see the full set of explanations for a chart and can color code each explanation by clicking in the palette attached to it. When a worker assigns the same color to multiple responses, the responses are pulled together in the list, creating a visual groupings. These groupings can act as intermediate clusters, allowing workers to visually compare similar comments side by side without having to rely as strongly on their working memory. However, because workers must still remember which colors they have assigned to which explanations, this approach may not scale well if the set of explanation is too large to be contained on a single screen.

Clustering explanations is a subjective task and the boundaries between clusters can vary depending on subtle interpretations of the explanation text. As a result, multiple workers – even well-intentioned and informed ones – may produce different clusterings. On one hand, this makes it practically impossible to pick a clustering by a simple majority vote. On the other hand, it is difficult to combine the clusterings produced by multiple workers. We observe that if multiple workers cluster explanations in a similar way, there is a high likelihood that the clustering is a good one. As a result, we assume that the clusterings that are the most dissimilar from all other clusterings of that set of explanations are likely to be bad, while the clusterings that are the most similar to all the others are likely to be good.

To determine which clusterings are the most similar to the others, we can compute the similarity between all of the clusterings of a chart’s explanations produced by different workers. We then pick the clustering that, on average, is the most similar to all of the other clusterings for that chart, which we term the “central” clustering. In our implementation, we surface only this clustering to the analyst.

4.3.4 Clustering Experiments

To compare our three clustering techniques, we conducted an experiment in which we used each method (automated, semi-automated, manual) to cluster the 156 explanations produced by workers in our test deployment, and then compared them against expert clusterings.

In the semi-automated condition, we created a comparison task for each pair of explanations given for each of the 12 charts. We grouped tasks into batches of 20, and asked five unique workers to complete each batch. We then averaged the worker scores for each comparison and used k-means clustering to produce a final set of clusters.

In the manual condition, we asked 10 different workers to cluster the explanations for each of the 12 charts.

Because clustering is subjective and no objective “best” clustering exists, we compared the results against manual clusterings generated by three expert users (the authors).

There are a variety of different ways to assess the similarity between two clusterings [2]. We compare clusterings using the F-measure, a similarity metric that is tolerant to small errors on large clusters, but intolerant to bi-directional impurities. The F-measure similarity for two clusterings is reported on a 0-1 scale, where 1 indicates that the clusterings are identical and 0 indicates that they are completely dissimilar.

There are many ways of evaluating the similarity between clusterings. To elaborate on the difficulty in designing a meaningful metric over the space of clusterings we briefly list a few questions we had encountered in our work. Let G be the gold standard clustering, and consider the following cases:

1. Which should be closer to G, a clustering that refines G, or a clustering that is refined by G?
2. Which should be closer to G, a clustering that agrees with G on the larger clusters but not the smaller ones or vice versa (a clustering that agrees with G on the smaller clusters but not the larger ones)?
3. Which should be closer to G, a clustering that either refines or is refined by G, or a clustering that is neither?

The clustering similarity metric that seemed to be the most appropriate for our needs is the F-measure. We motivate the choice of F, in correlation with the questions posed above, by noting that

1. it is tolerant to small errors on large clusters...
2. it is intolerant to bi-directional impurities

A more thorough discussion of various clustering similarity metrics can be found for example in [2].

The F measure of a single cluster is the harmonic average of the precision and the recall, and the F measure of an entire clustering is the weighted average of the F measures of all the clusters. Formally, given two clusterings \( L \) and \( R \), their F measure is

\[
F(L, R) = \frac{1}{\sum_{i} \left( \frac{|L_i|}{n} \cdot \max_{j} F(L_i, R_j) \right)}
\]

where \( n \) is the total number of clustered elements, \( i \) ranges over the number of clusters in \( L \) and \( j \) ranges over the clusters in \( R \). The function \( F(L_i, R_j) \) is defines as

\[
F(L_i, R_j) = \frac{2 \cdot \text{Recall}(L_i, R_j) \cdot \text{Precision}(L_i, R_j)}{\text{Recall}(L_i, R_j) + \text{Precision}(L_i, R_j)}
\]

where

\[
\text{Recall}(L_i, R_j) = \frac{|R_j \cap L_i|}{|L_i|}
\]

\[
\text{Precision}(L_i, R_j) = \frac{|R_j \cap L_i|}{|R_j|}
\]

4.3.5 Results

We compare the results of our experiments to the experts clusterings in (Figure 6).

While clustering can be a subjective process, the three experts performed quite consistently across most of the charts. Averaged across all charts, the three experts’ clusterings were very similar (F=0.84) and pairwise comparisons between the experts (E1-E2: F=0.85, E1-E3: F=0.85, E2-E3: F=0.83) revealed that no one expert was an outlier.

While the automated techniques performed as poorly as the average of crowd workers, we find that the “central” worker performed almost as similar to one of the experts as the experts were similar to each other.

5 The Explanation Management Interface

Our colleagues in this work have designed and implemented an expressive interface for presenting the mined information to the analyst. Figures 7 and 8 illustrate this interface and the ways in which it exposes information about explanation clusters and provenance.

6 Discussion

Here we offer a few observations based on our experience collecting, clustering, and exploring crowdsourced explanations, and offer ideas for future work.

Choice of clustering algorithms. The clustering task that we investigated in the paper is of a particular interesting size. Clustering ten short and noisy pieces of text is a task too small for most automatic statistical methods. This problem manifested itself in the poor results we achieved using all of our automated methods.

Conversely, clustering ten examples is nearing the limits of what humans can comfortably fit in their working memory and perform easily. This results in the observed variation in goodness of the manual clusterings collected from the crowd. More broadly, this line that straddles the sized of data that are easy for machines versus easy for humans to deal with is an interesting one, and it remains an open question when exactly should one switch from using one method to another.

Explanation segmentation. Our current implementation asks workers to separate distinct explanations into separate fields in the explanation microtask and allows them to select different source text for each. However, in practice, many workers still give multiple candidate explanations as part of a single paragraph or sentence. This can make responses difficult to group, since a single response may contain two or more explanations that belong in disparate clusters.

Workers in intermediate segmentation microtasks could break apart compound responses into their constituent explanations, but this introduces the potential for information or intent to be lost as workers break apart or alter others’ explanations.

This speaks to the broader issue of task granularity when crowdsourcing creative tasks. Breaking tasks into small, modular components makes it easier to compose tasks together and process results systematically. Small, straightforward tasks also reduce the potential for worker error, and make it easier to identify and discard poor results. However, small, segmented tasks may actually inhibit contributions from talented or knowledgeable workers, since they are not free to explore, author, or contribute outside the constraints of the task and cannot bring their expertise to bear on areas of the problem where it might be beneficial.

Cred composition. Our approach assumes a crowd composed largely of non-expert workers whose responses may be of variable quality - for example, workers recruited in online task markets like Mechanical Turk. With these workers in mind, we designed the analysis microtasks to be simple and included little interactivity. We also included quality-rating and source-checking analysis microtasks to help filter out low quality results and help analysts identify the most likely explanations.

However, more complex analyses or datasets that require specific domain knowledge may call for the use private crowds. We believe a similar workflow could be used to systematically collect and integrate findings from large crowds of trusted workers. In trusted crowds some quality-control mechanisms could be relaxed, reducing the number of post-processing steps and giving workers more freedom to explore. For example, trusted workers could be given the freedom to manipulate the visualization and explore alternate views of the dataset that might inform their explanations. Trusted workers could also self-
assess the quality of their explanations and sources, reducing the number of steps in the workflow while still providing metadata that analysts can use to filter and reorganize their results.

**Cross-view clustering.** Finally, we currently only cluster redundant explanations that were given for the same outlier. However, knowing that the same explanation was given for two or more outliers can also be helpful during analysis. Redundant explanations that span views may indicate a connection between the outliers or suggest an underlying phenomenon responsible for broader trends in the dataset. Exposing this sort of cross-view redundancy might help analysts use worker’s specific observations to make higher-level conclusions about the data.

7 Conclusion

In this paper we have explored several techniques that help analysts make sense of the explanations generated by workers in crowdsourced data analysis workflows. We show that on small scale tasks the average worker as well as automated NLP tools perform poorly. To overcome the shortage of the manual approach, we present a heuristic for identifying workers who produce high quality clusterings. To overcome the shortage of the automatic approach we propose collecting much more data and validating its quality. Additionally, we begin to identify the provenance of explanations generated by crowd workers, although much remains to be done in this area in future work. Finally, we present an interface that surfaces information to the analyst, that may help her make informed choices about the data she is analyzing.

Acknowledgments

Part of this text has been written and edited by Wesley Willett, Maneesh Agrawala and Björn Hartmann. The authors wish to thank them for their instruction throughout the work on this project.

References