Solving Mysteries with Crowds: Supporting Crowdsourced Sensemaking with a Modularized Pipeline and Context Slices

Tianyi Li

Dissertation submitted to the Faculty of the

Virginia Polytechnic Institute and State University

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in

Computer Science and Applications

Christopher L North, Chair

Kurt Luther, Co-chair

Gang Wang

Andrea L Kavanaugh

Gregorio Convertino

June 23, 2020

Blacksburg, Virginia

Keywords: Sensemaking, Text Analytics, Intelligence Analysis, Mystery Solving, Investigation, Crowdsourcing

Solving Mysteries with Crowds: Supporting Crowdsourced Sensemaking with a Modularized Pipeline and Context Slices

Tianyi Li

(ABSTRACT)

The increasing volume and complexity of text data are challenging the cognitive capabilities of expert analysts. Machine learning and crowdsourcing present new opportunities for large-scale sensemaking, but it remains a challenge to model the overall process so that many distributed agents can contribute to suitable components asynchronously and meaningfully. In this work, I explore how to crowdsource sensemaking for intelligence analysis. Specifically, I focus on the complex processes that include developing hypotheses from a raw dataset and iteratively refining the analysis. I first developed *Connect the Dots*, a web application that enables novice crowds to build relationship networks for exploratory analysis with "context slices". Then I developed *CrowdIA*, a software platform that implements a holistic crowd sensemaking pipeline to enable unsupervised crowd sensemaking. Using the pipeline as a testbed, I probed the errors and bottlenecks in crowdsourced sensemaking, and suggested design recommendations for integrated crowdsourcing systems. Building on these insights and to support iterative crowd sensemaking, I developed the concept of "crowd auditing" in which an auditor examines a pipeline of crowd analyses and diagnoses the problems to steer future refinement. I explored the design space and developed a crowd auditing tool, *CrowdTrace*. The core contributions of this work include a pipeline that enables distributed crowd collaboration to holistic sensemaking processes, two novel concepts of "context slices" and "crowd auditing", web applications that support crowd sensemaking and auditing, as well as design implications for crowd sensemaking systems. The hope is that the crowd sensemaking pipeline can serve to accelerate research on sensemaking, and contribute to helping people conduct in-depth investigations of large collections of information.

Solving Mysteries with Crowds: Supporting Crowdsourced Sensemaking with a Modularized Pipeline and Context Slices

Tianyi Li

(GENERAL AUDIENCE ABSTRACT)

In today's world, we have access to large amounts of data that provide opportunities to solve problems at unprecedented depths and scales. While machine learning offers powerful capabilities to support data analysis, to extract meaning from raw data is cognitively demanding and requires significant person-power. Crowdsourcing aggregates human intelligence, yet it remains a challenge for many distributed agents to collaborate asynchronously and meaningfully.

The contribution of this work is to explore how to use crowdsourcing to make sense of the copious and complex data. I first implemented the concept of "context slices", which split up complex sensemaking tasks by context, to support meaningful division of work. I developed a web application, *Connect the Dots*, which generates relationship networks from text documents with crowdsourcing and context slices. Then I developed a crowd sensemaking pipeline based on the expert sensemaking process. I implemented the pipeline as a web platform, *CrowdIA*, which guides crowds to solve mysteries without expert intervention. Using the pipeline as a testbed, I probed the errors and bottlenecks in crowd sensemaking and provided design recommendations for crowd intelligence systems. Finally, I introduced the concept of "crowd auditing", in which an auditor examines a pipeline of crowd analyses and diagnoses the problems to steer a top-down path of the pipeline and refine the crowd analysis. The hope is that the crowd sensemaking pipeline can serve to accelerate research on sensemaking, and contribute to helping people conduct in-depth investigations of large collections of data.

Acknowledgments

The past five years have been a transformative journey. I was extremely lucky to have had an amazing group of mentors, colleagues, friends, and family. I am deeply grateful to the following people, and many others not specifically named, without whom I would not have accomplished the work described in the following pages.

First, I'd like to thank my family. Despite the reluctance to separate from their only child, my parents have been extremely supportive for my pursuit of research passions in a different country. Their unconditional love, care, and trust have helped me through the ups and downs in the past five years.

My advisors, Dr. Chris North and Dr. Kurt Luther, have offered me the most inspiring and wholehearted mentorship. I would not have survived the steep learning curve without their tremendous support, guidance, trust, and patience. From the research mindset to paper writing, from methodologies to debugging, they enthusiastically and generously helped me and trained me from all levels. I am extremely blessed to have had two stellar and respectable scholars to introduce me to research and mentor me along the way. Dr. North and Dr. Luther are my trusted advisors and mentors not only in research but also in life. I would not have survived the stressful job search in this COVID-19 situation without their support. They embody the many qualities I admire as a scholar and a person that I want to be. Deeply grateful for what I received from them, I aim to become a professor like them and pass the spirit on to future generations of students.

My mentor Dr. Gregorio Convertino introduced me to industry research related yet beyond my dissertation topic. His open-mindedness, persistence, diligence, and modesty deeply influenced my life and work ethic. I cannot forget the countless heated discussions and

brainstorming, and how much I learned through apprenticeship. Gregorio helped me realized more potential of myself and I would not have developed my versatile research profile and interests without him.

My other dissertation committee members, Dr. Andrea Kavanaugh and Dr. Gang Wang, have provided continuous support throughout my milestones. They were always quick to provide feedback and encouragement to help me focus on the questions that matter, as well as raising interesting future work directions that are exciting and mind-opening.

It was a blessing for me to meet my mentor Dr. Mihaela Vorvoreanu in the last year of my Ph.D. Mickey introduced me to Microsoft Research, where I met and collaborated with many outstanding researchers. Mickey wielded her formidable talent as a public speaker, communicator, and researcher, to improve my various skills as a scholar. She also has been strong support during my stressful job search both professionally and emotionally.

I also want to thank my colleagues Yasmine Belghith, Chandler Manns, Yali Bian, Maoyuan Sun, Asmita Shah, Edward McEnrue, Jazmine Zurita, and Chris Lai, who have helped me through different stages and some of the most important ideas in this dissertation. Our wonderfully vibrant discussions and their fresh perspectives pushed the boundaries of my research far beyond anywhere I could go alone.

My lab mates, from both the InfoViz Lab and the Crowd Lab, have also been supporting me during the past five years. They were constantly available with a sympathetic ear and clever research insights. Their valuable input in my pilot studies and feedback for my practice talks helped me present my work to the broader research community.

The administrative and technical staff at Virginia Tech, especially Sharon Kinder-Potter, Melanie Darden and Teresa Hall, made my life in graduate school easier in countless ways.

I would also like to thank my partner and dear friends for their continuous support and encouragement. Their endless reserves of enthusiasm helped to replenish my own when it mattered most.

Last but not least, this research was funded in part by NSF Grants IIS-1527453, IIS-1651969, and IIS-1447416.

Contents

List of Figures xv

List of Tables xx

1 Introduction 1

1.1 Domain and Scope: Text Data and Hidden Plots 1

1.2 Background and Motivation: Sensemaking and the Wisdom of Crowds . . . 2

1.3 Challenges and Goals 4

1.4 Major Research Questions 6

1.4.1 RQ 1: How to Enable Crowds to Extract Hidden Connections and Produce a Holistic View of the Information in Text Documents with Local Views of the Data? 7

1.4.2 RQ 2: How to Establish a Bottom-up Pipeline to Enable Many Distributed Agents to Develop a Theory from the Raw Data? 8

1.4.3 RQ 3: What are the Limitations and Challenges in the Bottom-up Pipeline for Crowd Sensemaking? 9

1.4.4 RQ 4: How to Refine the Crowdsourced Analysis with a Top-down Process? 10

1.5 Structure of This Dissertation 12

1.5.1 Complete List of Research Questions 13

2 Review of Literature **15**

2.1 Sensemaking Loop 15

2.2 Collaborative Sensemaking by Experts and Groups 16

2.2.1 Challenges for Individual Experts 16

2.2.2 Additional Needs for Shared Artifacts and Common Ground 18

2.2.3 Hand-off Timing and Instruments for Asynchronous Collaboration . . 19

2.2.4 Teammate Inaccuracy Blindness and Reluctance to Share Information. 20

2.3 Crowdsourcing Complex Cognitive Tasks: Large-Scale Coordination 21

2.3.1 Expert Intervention to Prepare and Guide the Crowd Tasks 22

2.3.2 Crowd Sensemaking Workflows and Paradigms 23

2.4 Quality Control of Crowdsourced Analysis 25

2.4.1 Crowd Work Quality and Influencing Factors 25

2.4.2 Quality Control Challenges for Crowdsourced Sensemaking 26

2.4.3 Coordination Artifacts for Complex Work: Trade-offs between Predefined Guidance and Situated Adaptation 28

2.4.4 Techniques and Best Practices for Better Crowd Outcome 29

2.5 Providing Feedback in Sensemaking 33

3 Context Slices and Crowdsourced Relationship Graph Building **35**

3.1 Connect the Dots: System Description 35
3.1.1 Document View 36
3.1.2 Connection Workspace 37
3.2 Study Design 38
3.2.1 Dataset 38
3.2.2 Participants 40
3.2.3 Procedure 40
3.2.4 Data Collection 41
3.3 Results 41
3.3.1 RQ 1.1: Types of Connections 42
3.3.2 RQ 1.2: Comparing Slicing Methods 44
3.3.3 RQ 1.3: Finding Key Connections 46
3.3.4 RQ 1.4: Prioritizing Important Entities 50
3.4 Discussion 52
3.4.1 Context Slicing with Overlapping Entities 53
3.4.2 Coverage of Gold Standard with Thresholds 53
3.4.3 Sources of Strategic Information Retrieval 54
3.4.4 Meaningfulness of Crowd Connections 56
3.4.5 Limitations and Future Work 57
3.4.6 Lessons Learned 57

4 CrowdIA Pipeline: Bottom-up Building Path **59**

4.1 Design Process: Preliminary Studies 60

4.1.1 RQ 2.1: Identifying Step Inputs and Outputs with Individual Participants 60

4.1.2 RQ 2.2: Distributing Input and Aggregating Output with Crowd Workers 62

4.2 The CrowdIA System 64

4.2.1 Implementation 64

4.2.2 Pipeline Structure and Step Definition 66

4.2.3 Step 1: Search and Filter 67

4.2.4 Step 2: Read and Extract 67

4.2.5 Step 3: Schematize 68

4.2.6 Step 4: Build Case 69

4.2.7 Step 5: Tell Story 71

4.2.8 Refining Path: Top-Down 71

4.3 Evaluation: Solving Mysteries with Crowds 72

4.3.1 Method 73

4.3.2 Results of Easy Dataset 73

4.3.3 Results of Moderate Dataset 74

4.3.4 Results of Difficult Dataset 75

4.4 Discussion 79

4.4.1 RQ 2.1: How can we modularize the sensemaking process? 79

4.4.2 RQ 2.2: How do we distribute and aggregate the analysis in each step? 81

4.4.3 RQ 2.3: How do crowds perform in solving mysteries with the modularized pipeline? 83

4.4.4 Generalizability 84

4.4.5 Limitations and Future Work 86

5 Challenges in the Bottom-up Pipeline **88**

5.1 Methods 88

5.1.1 Participants 88

5.1.2 Task and Procedure 89

5.1.3 Context Slicing Methods and Choices. 89

5.1.4 Pipeline Execution Walkthrough 90

5.1.5 Data Analysis 92

5.1.6 Limitations 94

5.2 Results 94

5.2.1 RQ 3.1: Error Types and Propagation 95

5.2.2 RQ 3.2: Impact of Context 102

5.3 Discussion and Design Implications 108

5.3.1 Error Propagation Among Crowds: Easier Hand-off but More Inaccuracy Blindness. 108

5.3.2 Design Recommendations for Each Step 109

5.3.3 Generalizability and Future Work 112

6 CrowdIA Pipeline: Top-down Refining Path 117

6.1 Preliminary Studies (RQ 4.1) 117

6.1.1 Can Crowds Refine Existing Analyses Directly? 118

6.1.2 Can Crowds Fix Identified Problems? 119

6.1.3 Can Individuals Identify Problems in Crowd Analysis? 120

6.1.4 Design Goals: Supporting Crowd Auditing 121

6.2 CrowdTrace (RQ 4.2) 123

6.2.1 Overview 123

6.2.2 Identifying Problems 124

6.2.3 Providing Feedback 126

6.2.4 Example User Scenario 128

6.3 Evaluation Study Design 129

6.3.1 Performance Metrics 130

6.3.2 Participants 131

6.3.3 Procedure 131

6.4 Results 132

6.4.1 Performance of Crowd Auditing 133

6.4.2 Performance of Creating Microtasks 135

6.4.3 Audit Strategies 136

6.4.4 Efficiency and Learnability: Time Measurement 139

6.4.5 Efficiency and Learnability: Survey Response 141

6.5 Discussion 143

6.5.1 Challenges and opportunities in Crowd Auditing 143

6.5.2 Design Implications for Scaffolding Crowd Auditing 145

6.5.3 Generalizability and Broader Impacts 148

7 Conclusion **151**

7.1 Addressing the Research Questions 151

7.1.1 RQ 1: How to Enable Crowds to Extract Hidden Connections and Produce a Holistic View of the Information in Text Documents with Local Views of the Data? 151

7.1.2 RQ 2: How to Establish a Bottom-up Pipeline to Enable Many Distributed Agents to Develop a Theory from the Raw Data? 154

7.1.3 RQ 3: What are the Limitations and Challenges in the Bottom-up Pipeline for Crowd Sensemaking? 156

7.1.4 RQ 4: How to Refine the Crowdsourced Analysis with a Top-down Process? 158

7.2 Contributions and Broader Implications 161

7.3 Future Work 162

7.3.1 Re-examining the Design Decisions 163

7.3.2 Mixed-Initiative Intelligent Systems 166

Bibliography **169**

Appendices **205**

Appendix A Appendix for Chapter 4 **206**

A.1 Datasets 206

A.2 Crowd Analysis of Easy Dataset 209

A.3 Crowd Analysis of Moderate Dataset 211

A.3.1 Additional Experiment Results of Moderate Dataset 212

A.4 Assumptions 215

A.4.1 Assumptions about External Data Resources 217

A.4.2 Assumptions about Reportable Results 217

Appendix B Appendix for Chapter 5 **219**

B.1 Gold standard analysis and decision rationales 219

B.2 Error propagation shown in diagrams 222

List of Figures

2.1 Sensemaking loop [153], image source [44] (left). Example of how each component is modularized in the pipeline (right). 15

3.1 The Connect the Dots web application interface 36

3.2 Example subgraph of connections made by five crowd workers for one context slice. 42

3.3 Average number of connections per slice, and multi-document node pairs in different slicing methods. 45

3.4 Precision, recall, and f-measure values for varying worker vote thresholds. . . 47

3.5 Relationship graphs of person names by document co-occurrence, gold standard, and crowd workers. 48

3.6 Ranked entity pairs by the number of workers connecting them. Gold lines are entity pairs from the gold standard. Blue lines are other possible entity pairs. 51

3.7 Rank entities by their degrees in the graph. 52

4.1 Prototype Interface: Example task specification interface, input data file, and output answer sheet. 61

4.2 Intermediate analysis results by individual participants in Steps 1 and 2. Blue bars are their initial results, green bars are their second path edits, and orange bars are correct answers contained in each participant's results. 61

4.3 Example schemas created by individual participants (left) and crowd workers (right) in the preliminary studies. 63

4.4 Example crowd worker interface for Step 4. On the top are task instructions including the global context (first line), task overview (first paragraph), and action items (bullet points). On the bottom left is one context slice as local task input (available material). On the bottom right is the local task output where crowds fill in and submit their analysis. 65

4.5 Modularized sensemaking pipeline. Step 1 searches external data sources for relevant documents. Step 2 extracts important information pieces from the relevant documents. Step 3 organizes information pieces into profile schemas. Step 4 compares and merges schemas to develop hypotheses. Step 5 synthesizes the best hypotheses as the final presentation. 65

4.6 Step 1: Search and Filter. Crowds independently rate document relevance from 0 (completely irrelevant) to 100 (completely relevant). Using a predefined threshold, each relevance rating is converted to a binary vote. Documents with the majority vote will be passed to Step 2. 67

4.7 Step 2: Read and Extract. CrowdIA groups documents with overlapping entities into context slices of size $n = 2$ (A). The first batch of crowd workers extracts information pieces from context slices (B). The information pieces are then regrouped by their source documents into new context slices (C). The following batches of crowds review information pieces (D). The process continues until no new revisions are made. 68

4.8 Step 3: Schematize. Crowds identify potential target locations and tag the information pieces with known elements. Information pieces are tagged with tags that earned the crowd's majority vote and organized into profiles of the candidate targets. 70

4.9 Step 4: Build Case. Crowds compare candidate profiles and merge aliases. As in a single-elimination competition, workers in Step 4 rank candidates by their perceived likelihood of being the target location. 70

4.10 Step 5: Tell Story. Crowds put together the information in the winning profile and write a complete narrative. The presentation is ready when no new revisions are made. 71

5.1 Experiment Design: Besides the five different steps in the pipeline, we manipulate the quality of step input (gold-standard or crowd-generated) and the size of context slices (1, 3, or all items in the step inputs). There are 4 conditions in total. 89

5.2 Task performance measured by two sources of errors: data quality and task behaviors 93

5.3 RQ 3.1: Errors in GI (gold-standard input) and CI (crowd-generated input) conditions. 102

5.4 RQ 3.2: Errors in uni-item, triple-item and all-item conditions. 103

5.5 Time spent on each step when given different amount of local context 114

6.1 Example of crowd comments in the first preliminary study. 119

6.2 Problem description table for participants to fill in and example results by participants. 121

6.3 CrowdTrace. Auditing interface for identifying problems, ... 123

6.4 Microtask creation interface and example microtask created by a participant. 127

6.5 The stacked bars show the number of important and other problems identified by each participant, sorted by the number of important problems; the lines show the number of annotations and microtasks created by each participant. 133

6.6 The number of participants who successfully identified each problem, sorted by frequency. 134

6.7 A typology of audit strategies used and externalized by participants in crowd auditing. 137

6.8 Time management of each auditor. Orange bars are time spent on creating annotations, and olive green bars are time spent on creating microtasks. . . 138

6.9 Left: Time elapsed before annotation (minutes) decreases as more annotations are made. Right: Time spent on creating each microtask decreases as more microtasks are created. 140

6.10 Questions and responses in the post survey. Individual responses are shown in dots. 141

A.1 Single elimination competition of profiles in Step 4. 214

B.1 Error Propagation in Uni-item Condition: the pink colored items are irrelevant documents, info pieces, and location profiles; the green colored ones are relevant to solving the mystery. 222

B.2 Error propagation in triple-item condition 222

B.3 Error propagation in all-item condition 223

List of Tables

1.1 Challenges and goals 5

4.1 Customized context slicing of each step depends on the level of analysis and the goal of the step. 82

5.1 Number of workers hired in each step and each condition, and the total number of workers in each step across conditions. While Step 5 only requires two workers (one writer and one reviewer), for the purposes of this chapter, we ran it 4 more times ($4 \times 2 = 8$ workers) to gather more data and be comparable to the other conditions. 95

A.1 Easy dataset adapted from a brain teaser. The correct answer is that Serina is the culprit. 207

A.2 Moderate dataset skeleton adapted from the card game Clue. 208

A.3 Example documents from the difficult dataset adapted from *The Sign of the Crescent*. 208

A.4 Profiles generated from information pieces tagged by crowds in Step 3. . . . 213

A.5 Additional experiments: Profiles generated by information pieces tagged by crowds in Step 3. 216

Chapter 1

Introduction

Intelligence analysts working to prevent terrorist attacks and preserve national security have access to an unprecedented wealth of data about persons of interest. Yet, events such as the September 11th, 2001 terrorist attacks and the miscall on weapons of mass destruction in Iraq — "the two major U.S. intelligence failures of this century" [43] — illustrate the difficulties that even experienced professionals face in analyzing this data and the high-stakes consequences of failure. Crowdsourcing presents new opportunities to manage this challenge by augmenting the cognitive work of individuals, providing richer analysis than automated approaches and scaling better than traditional intelligence work [28, 36, 81]. However, this requires finding a way for many distributed novice workers to contribute meaningfully, through small and independent tasks, to the sensemaking process of experts.

1.1 Domain and Scope: Text Data and Hidden Plots

Modern technologies such as social media and mobile devices produce a growing wealth of data. Such data offers an unprecedented opportunity to develop a deeper and more global view of the world, but to make sense of such data is challenging. Intelligence analysis, for example, faces the ongoing challenges of distinguishing crucial information from noise and dealing with incomplete pieces. Marshaling and synthesizing heaps of evidence is especially difficult. As observed by Wright et al., "To get the big picture by looking at many pages

of text, the analyst relies heavily on memory to connect the dots" [188]. Failing to make sense of the available data could result in spreading misinformation and exacerbating biases in online communities [139]. Worse still, failing to prevent terrorist attacks or solve crimes could harm national security. Other similar scenarios include sentiment analysis of product reviews and social media contents, natural language understanding for AI-infused systems and conversational interfaces, and so on. In this dissertation, I focus on a class of problems that involve solving mysteries, in which analysts must sort through many snippets of textual information to identify a latent plot, such as a suspect in a murder case or the target location of a terrorist attack.

1.2 Background and Motivation: Sensemaking and the Wisdom of Crowds

Sensemaking offers great potential to understand the meaning and patterns contained within large quantities of unstructured, noisy source materials. Sensemaking is used in many domains, from intelligence analysis to investigative journalism. Pirolli and Card modeled the expert sensemaking process as an iterative loop with multiple interdependent steps that involves foraging for relevant information and synthesizing it into hypotheses [153]. The process ultimately relies on the experts to make ad hoc decisions on which steps to conduct and how to arrange the workflow. Managing this complex process is cognitively demanding and often requires significant person-power [189]. Making sense of massive amounts of complex information that comes from various sources, and discerning critical patterns and anomalies has always been challenging for individual experts.

Several areas of research attempt to support this sensemaking process. Visual analytics

tools have been developed to leverage technological support for some specific steps [58, 71]. Collaborative sensemaking among experts can bring together diverse expertise and perspectives but often suffers from biases and inefficiencies [68, 166]. Machine learning can process large amount of data and provide starting points for analysts [190], but deciphering the rich information encoded in text data is still AI-hard [194].

Alternatively, crowdsourcing is a powerful new paradigm that augments distributed human intelligence at a large scale. The crowd intelligence shows potential to bridge the gap between the information overload and the limited cognitive capacity of individual experts. Online voluntary communities have led to remarkable successes in solving a diversity of difficult problems ranging from specific software develop questions [135, 167] to open mathematical research questions [45]. However, the misidentification of the Boston Marathon bomber by the Reddit crowds [117] cautions that without some workflows or central coordination, the crowd discussions can be heavily biased by early, even arbitrarily chosen opinions [144]. Some researchers introduce role- and team-based hierarchy to organize an expert crowd in not only the work but also the continuous coordination and adaptation of the division of labor and workflow [175]. Delegating these meta-decisions to the crowds can efficiently coordinate on-demand workers and make prompt and situated adaptation in the workflows based on the data and the work progress. Another booming category of crowdsourcing tools is interactive systems that allow requesters to communicate with workers, clarify task specifications, and provide feedback in real time [12, 115]. The direct and natural cooperation reduced the limitations of micro crowdsourcing tasks and workflows, thus expanding crowdsourcing techniques to a broader application. Despite the flexibility, adapting the workflows in real time requires the coordinators, either some expert workers or the requester, to stay online the entire session and pay close attention to the work status and progress. Besides, expert crowd workers are more expensive to hire and some expertise is difficult to find.

Some crowdsourcing Internet marketplaces like Amazon Mechanical Turk (MTurk) provide a broader and more accessible pool of novice, transient crowd workers. By decomposing a big problem into many small, manageable problems and aggregating results from small solutions into a big meaningful result, prior research has successfully used crowds for complex data analysis tasks like taxonomy generation [34], bottom-up qualitative analysis [6], and organizing online information [67].

However, most novice crowd sensemaking solutions focus on well-defined sub-problems (e.g., schematizing text data [130]), provide crowds with ideal input data (e.g., raw documents manually broken down by researchers into smaller text items [35]), or require facilitation by experts [25]. The crowd results are perceived as useful and a good starting point, but usually require additional work by requesters [196]. Furthermore, distributed work comes with significant challenges for quality control. Mixed-quality work remains "one of the main roadblocks to having crowdsourcing achieve its full potential" [183]. Complex sensemaking that draws on multiple intermediate crowdsourcing processes incurs even more challenges, with errors and mistakes propagating from their source to later analyses [123].

In this work, I explore how to modularize the complex and open-ended sensemaking process for the novice, transient crowd workers to collaboratively solve mysteries, the errors and bottlenecks that challenges crowd performance, as well as how to refine the crowd analysis to support iterative refinement in crowdsourced sensemaking.

1.3 Challenges and Goals

To enable crowd collaboration in sensemaking with large amounts of data, we first need to address two major problems for modularization. The first problem is **modularizing the data**. Sensemaking requires a holistic view of the data, making it difficult to subdivide the

Table 1.1: Challenges and goals

Challenge	**Goal**
C1: Holistic view of the data vs. distributed local contribution	G1: Modularize the data (context slices)
C2: Highly integrated cognitive activities vs. large-scale collaboration	G2: Modularize the process (pipeline)
C3: Mixed-quality crowd analysis and error propagation	G3: Model the errors and bottlenecks
C4: Data-specific and interdependent crowd errors make iteration on the analysis difficult	G4: Refine multiple interdependent steps of crowd analysis

data into small local slices while preserving global data context for crowd workers [177]. The second problem is **modularizing the process**. The entire sensemaking process required for solving mysteries is a highly integrated cognitive activity composed of iterative information foraging, schematizing, and synthesizing, which is difficult to formalize into one single workflow of microtasks for novice crowd workers [170]. The two problems intersect and cannot be solved independently in the crowdsourcing setting. For each modularized sub-process, the data should also be modularized for individual crowd workers to work on. To overcome these problems, we need a model that adequately captures and translates the expert sensemaking process for large crowds of transient novice workers, and distribute the data for each individual to analyze.

Furthermore, unsupervised crowd sensemaking, where crowd analyses are directly handed off to another group of workers for the next step of the analysis, is subject to **errors and bottlenecks**. First, the crowd makes mistakes in each step, which could compound when propagated down the pipeline, potentially causing an incorrect final outcome. Second, in addition to traditional quality control challenges in crowdsourced work (such as task design), the quality of pipelined crowd analysis is also influenced by how the data in each step is distributed among microtasks, the sensemaking challenges in each step, as well as the crowd

performance in previous steps. Understanding these effects is important for improving the crowd analysis outcome and designing more robust crowdsourced sensemaking pipelines.

Therefore, **improving the quality of a pipeline of crowd analysis** requires more than enhancing the crowd performance in a single step. In-depth evaluations of each step might not be optimal when there are multiple interdependent steps of crowd processes, especially that crowd errors in a step could be inherited from previous crowd results. Given the interconnected structure of the sensemaking steps, deciding where and how to fix the errors also requires careful consideration.

1.4 Major Research Questions

To tackle these challenges, I focus this dissertation on exploring the following major research questions:

RQ1. How to enable crowds to extract hidden connections and produce a holistic view of the information in text documents with local views of the data?

RQ2. How to establish a bottom-up pipeline to enable many distributed agents to develop a theory from the raw data?

RQ3. What are the limitations and challenges in the bottom-up pipeline for crowd sensemaking?

RQ4. How to refine the crowdsourced analysis with a top-down process?

In the next sections, I discuss each of these research questions in detail.

1.4.1 RQ 1: How to Enable Crowds to Extract Hidden Connections and Produce a Holistic View of the Information in Text Documents with Local Views of the Data?

One important task of sensemaking processes is to construct a big picture of the distributed and hidden information in the data, i.e. "connect the dots". This includes the iterative sub-processes of extracting and schematizing information, connecting the information foraging and synthesizing processes in the sensemaking loop [154]. The requirement of a holistic view of the data poses the first challenge for crowdsourced sensemaking. In the first milestone, I investigate the following research sub-questions.

RQ 1.1: What types of connections does the crowd create?

RQ 1.2: How do different slicing methods influence the crowd results?

RQ 1.3: When using context slices, how well can crowds find the connections needed for the solution?

RQ 1.4: When using context slices, how can we distinguish or prioritize the most important entities?

To address this "big picture" challenge, we introduce a novel concept called *context slices*, in which datasets are restructured to enable in-depth inquiry by transient novice crowd workers. Context slices decompose the dataset into smaller subsets that are appropriate for one crowd worker to analyze in a micro-task. To explore the utility of context slices and investigate the four research sub-questions, I developed a web application, *Connect the Dots*, that helps crowd workers make connections to elicit entity relationships from text documents. The connections are then aggregated and visualized as a relationship network.

I conducted an experiment in which 275 paid crowd workers used this software to create connections under 3 conditions, each using a different slicing method. The crowd generated nearly 6000 connections from documents about a fictional terrorist plot. With context slices, crowds successfully extracted most of the connections that analysts would need, along with accurate and meaningful descriptions. The connections constitute a relationship network that can be used to retrieve and schematize information from the source documents. I will present more details in Chapter 3.

1.4.2 RQ 2: How to Establish a Bottom-up Pipeline to Enable Many Distributed Agents to Develop a Theory from the Raw Data?

Crowdsourcing has shown success in multiple sensemaking steps but with important caveats and assumptions, namely: 1) the step input is perfect, i.e. nothing in the step input would hinder the analysis in the current step; and 2) the imperfect step output will be curated and managed by some experts or researchers. My work to address RQ 1 successfully enabled crowd collaboration on the connecting steps between the information foraging and synthesizing. Moving forward, can we connect multiple crowdsourced sensemaking steps? How can we release these assumptions and crowdsource the entire sensemaking process? Context slices have been proved to be helpful for some sensemaking steps, how can we apply this concept in other steps?

In this milestone, I study how to modularize the process and the data in complex sensemaking so that large-scale, distributed agents can meaningfully contribute to suitable components. Starting from the copious external data source, we first need to construct a bottom-up building path of the pipeline to develop a theory from the raw data. To this end, I aim to

address the following research questions:

RQ 2.1: To support crowds, how can we formally modularize the sensemaking process into a series of steps that each defines the information needs (*Step Input*) and intermediate analysis results (*Step Output*)?

RQ 2.2: Within each step, how do we slice the *Step Input* into contextualized microtasks for individual crowd workers, and aggregate the local analysis results into *Step Output*?

RQ 2.3: How well do crowds perform in solving mysteries with the modularized sensemaking process, and specifically, how do crowds perform in each step?

To address the first two research questions, I conducted two preliminary studies with both individual users and crowds. Informed by the study results, I modularize the bottom-up sensemaking process into a pipeline of clearly defined steps and designed the context slicing methods for each step. I implemented the pipeline as a software platform called *CrowdIA*. To address RQ 2.3, I evaluated the pipeline by deploying CrowdIA on Amazon Mechanical Turk (MTurk) to guide crowds in solving three mysteries. In these empirical studies, the crowds successfully solved easy and moderate mysteries and were one step away from solving a difficult mystery. I will present more details in Chapter 4.

1.4.3 RQ 3: What are the Limitations and Challenges in the Bottom-up Pipeline for Crowd Sensemaking?

The pipeline connects different steps of crowd sensemaking and passes the crowd results from one step to another, including their errors and mistakes. When the entire sensemaking process is crowdsourced, will the same challenges for experts prohibit the crowd analysis? Does the crowd do better in the face of some challenges, and what new issues do they face?

Prior crowdsourcing research studied crowd performance in focused, well-defined tasks. In this milestone, I aim for a deeper understanding of the challenges in crowdsourced sensemaking in a holistic process with different types of tasks and stages of analysis. I explore the following research questions:

RQ 3.1: What are the errors (type and frequency) workers make in a crowdsourced sensemaking pipeline, both within each step and across steps?

RQ 3.2: How does the amount of local data context affect the errors within and across steps in a crowdsourced sensemaking pipeline?

To answer the research questions, I conducted a series of mixed-method studies with 325 crowd workers to work on the difficult mystery that the crowds failed to solve in RQ 2. I first investigated how crowd performance is influenced when given crowd-generated input versus gold-standard input (RQ2.1). I then examined how the amount of local data context influences worker performance and error propagation (RQ2.2). I evaluated the crowd performance by comparing their analysis to a gold-standard analysis adapted from the dataset's answer sheet. I classified the types of errors that occurred in each intermediate step, specifically focusing on the source and impact of errors, and how the amount of local context influence the error propagation in the pipeline. I will present more details in Chapter 5.

1.4.4 RQ 4: How to Refine the Crowdsourced Analysis with a Top-down Process?

Sensemaking is "never fully complete" [38]. Crowdsourced sensemaking also needs to support the iterative refinement of the analysis. The discovered evidence sheds light on other seemingly irrelevant but actually crucial hints to progressively uncover the hidden plots in

mysteries. Therefore, it is not enough to chain different steps into a building path of the pipeline and improve the analysis locally at each step. We need a mechanism to organize and broadcast the new findings from the execution of the building path back to each step, to iteratively retrieve the missing pieces and calibrate the analysis process. With a deeper understanding of crowd performance in the building path from RQ 3, I explore the design of the refining path of the pipeline with the following research questions:

RQ 4.1: What are the key challenges in refining a pipeline of crowd analyses?

RQ 4.2: How can technology be designed to support the refinement of a pipelined crowd analysis?

To address these questions, I first explored the design space and investigate the main challenges in refining crowd analysis with a series of preliminary studies. Informed by the studies, I proposed a novel approach, *crowd auditing*, to address the challenge of refining mixed-quality results in a pipeline of crowd analyses. Crowd auditing emphasizes building on the insights gained in the existing analyses, probing the problems across different steps, and steering the refinement with a top-down approach. I developed *CrowdTrace*, a software prototype to support crowd auditing. CrowdTrace visualizes the structure and provenance of crowd analyses and provides support for identifying problems and creating microtasks. The system's goal is to help auditors in providing actionable feedback and steering crowdsourced refinement of the analyses, enabling iterative crowdsourced sensemaking.

We evaluated CrowdTrace in a user study under the scenario of refining crowd analyses of a fictional terrorist attack plot. With CrowdTrace, auditors successfully identified important problems in the crowd analyses and created high-quality microtasks to fix the problems. We observed that auditors became faster at crowd auditing over time. While auditors recognized that identifying problems and creating microtasks are both difficult, they considered

CrowdTrace helpful for both tasks, and found it beneficial to have crowd results available for the overall analysis outcome. I will present more details in Chapter 6

1.5 Structure of This Dissertation

Chapter 1 introduces the background and motivation of this work, lists the key challenges and goals, as well as the research questions. This section concludes Chapter 1. Chapter 2 presents a literature review, focusing on the sensemaking loop, the prior research on collaborative sensemaking, lessons, and limitations from previous successful applications of crowdsourcing in complex tasks, the quality control and credibility of crowdsourced analysis, and the trade-off between predefined guidance and situated adaptation in designing crowdsourcing workflows.

Chapter 3 reports on the development of the "context slices" concept and an empirical study to address Research Question 1. The *Connect the Dots* project focused on extracting the key relationship between the entities from text documents to support mystery solving. The outcome is a deeper understanding of the crowd's potential in conducting complicated and exploratory sensemaking and the design of context slices.

Chapter 4 addresses Research Question 2 by modularizing the overall sensemaking process and conduct context slicing in each step, presenting the design of the sensemaking pipeline and the CrowdIA system. This chapter also describes the design process, implementation of the CrowdIA system, and an evaluation study of the building path.

Chapter 5 addresses Research Question 3 with a systematic evaluation of the pipeline and context slices proposed in the previous chapters. I developed a typology of the errors and bottlenecks in crowdsourced sensemaking. This chapter also makes design recommendations

for integrated crowdsourcing systems.

In Chapter 6, I address Research Question 4 with a novel approach called "crowd auditing" to drive the top-down refining process of pipelined crowd analyses. This chapter presents the preliminary study that explores the design space, the CrowdTrace system, an evaluation study, and design implications for auditing crowdsourced sensemaking results.

In Chapter 7, I revisit my four research questions, summarize the main contributions, and speculate future research directions in crowdsourced sensemaking.

1.5.1 Complete List of Research Questions

RQ 1: How to enable crowds to extract hidden connections and produce a holistic view of the information in text documents with local views of the data?

- RQ 1.1: What types of connections does the crowd create?
- RQ 1.2: How do different slicing methods influence the crowd results?
- RQ 1.3: When using context slices, how well can crowds find the connections needed for the solution?
- RQ 1.4: When using context slices, how can we distinguish or prioritize the most important entities?

RQ 2: How to establish a bottom-up pipeline to enable many distributed agents to develop a theory from the raw data?

- RQ 2.1: To support crowds, how can we formally modularize the sensemaking process into a series of steps that each defines the information needs (*Step Input*) and intermediate analysis results (*Step Output*)?

RQ 2.2: Within each step, how do we slice the *Step Input* into contextualized microtasks for individual crowd workers, and aggregate the local analysis results into *Step Output*?

RQ 2.3: How well do crowds perform in solving mysteries with the modularized sensemaking process, and specifically, how do crowds perform in each step?

RQ 3: What are the limitations and challenges in the bottom-up pipeline for crowd sensemaking?

RQ 3.1: What are the errors (type and frequency) workers make in a crowdsourced sensemaking pipeline, both within each step and across steps?

RQ 3.2: How does the amount of local data context affect the errors within and across steps in a crowdsourced sensemaking pipeline?

RQ 4: How to refine the crowdsourced analysis with a top-down process?

RQ 4.1: What are the key challenges in refining a pipeline of crowd analyses?

RQ 4.2: How can technology be designed to support refinement of a pipelined crowd analysis?

Chapter 2

Review of Literature

2.1 Sensemaking Loop

Intelligence analysts make sense of large amounts of information by iteratively foraging for relevant source data (*1. search and filter*), extracting useful information (*2. read and extract*), organizing and re-representing the information with their mental models (*3. schematize*), developing hypotheses from different perspectives (*4. build case*), and deciding on the best explanation (*5. tell story*). Pirolli et al. [153] informally modelled this process as a main loop composed of an information foraging sub-loop and a synthesizing sub-loop, each iterating on smaller intertwined steps (Fig. 2.1 left). HCI research on improving sensemaking in different domains and settings, as we review in later sections, can be considered as fitting in different parts of the sensemaking loop.

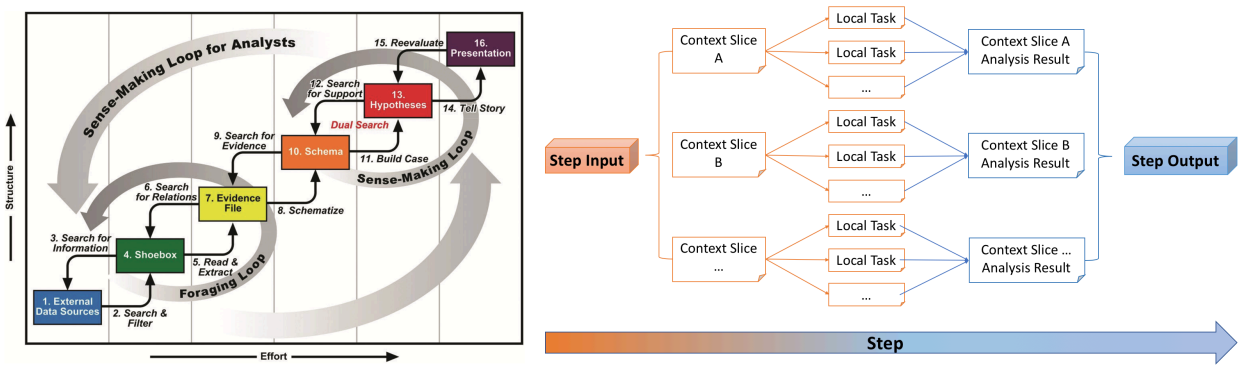

Figure 2.1: Sensemaking loop [153], image source [44] (left). Example of how each component is modularized in the pipeline (right).

2.2 Collaborative Sensemaking by Experts and Groups

Traditional intelligence analysts face the ongoing challenges of parsing, marshaling, and synthesizing large quantities of unstructured data. Analysts need to distinguish pertinent information from noise, deal with incomplete pieces, find potential suspects, to eventually identify the criminal or suspect [70, 189]. Typical errors of individual expert analysts include wrong or missing information due to inaccurate memory, misinterpreting evidence due to cognitive fatigue, and biases due to perception constraints [38, 86].

2.2.1 Challenges for Individual Experts

Making sense of a large amount of text and uncovering the hidden plots is challenging even for devoted analysts [39]. It is a time-consuming and cognitively demanding process that involves reading through a large amount of text, internalizing and mentally connecting the scattered information, developing hypotheses accordingly, and presenting the analysis results [153].

Mental Process One key challenge in sensemaking is to collect and mentally compare the relevant information scattered across many locations. The same mental processes can have different outcomes with different analysts. "Which information is attended to, how it is organized, and the meaning attributed to it all depends on the analyst's past experience, education, cultural values, role requirements, and organizational norms, as well as the specifics of the information received" [87].

When solving a more perplexing problem, analysts are challenged by data with higher quantity and complexity. The retention interval between encoding and retrieval of the memory lengthens, leading to an increase in both the amount and the likelihood of a memory error

or gap. More importantly, the relevance and usefulness of the information are often more obscure and relative to other information hidden in the data set. Not having access to the right context may prohibit the analysts in their judgment and direction of analysis [30].

We summarize the challenges in the overall sensemaking progress as follows. First, *perception limitation of mental models*. This includes different focus (Step 1 and 2), interpretations (Step 2, 4, 5), and organizations (Step 3) of information by different experts, as well as cognitive biases (including "evaluation of evidence, perception of cause and effect, estimating probabilities" [87]). Second, *memory limitations*. Individuals have a limit of memory storage and retrieval and could make mistakes and have gaps in their memories.

Context and Existing Knowledge Uncovering the hidden, unknown knowledge from text documents requires selecting the right parts and amount of existing knowledge as the context to connect the dots. A classical example is that "the veterinary pathologist Eugen Semmer (1870) discovered penicillin many years before Alexander Fleming but did not realize that he had done so" [46]. Although he discovered the connection between the unexpected recovery of the dying horses in his lab to the presence of spores of the fungus, he did not realize the importance of this finding and even eradicated the spores from his lab. On the other hand, Fleming, who also discovered penicillin accidentally, recognized the possible practical application of his results, which ended up saving countless lives.

For the mystery-solving problems that we focus on in this work, putting each information piece in the right context is important for uncovering the less obvious relationships. While we assume there is no need to use any domain knowledge (A.4) to solve the mysteries, it would require applying the common sense to connect distributed facts appropriately. For example, it is perfectly normal that a guy named Mark Davis works on the vending machines in New York Stock Exchange. Such information might not be extracted as important evidence

when solving a mystery about a fictional terrorist attack. However, if put together with another fact that a terrorist named Hamid Alwan used a fake passport under the name Mark Davis, the previous information piece now becomes more relevant. For one thing, the two information pieces describe two different facts about the name Mark Davis. For another, the common sense that fake passport and name used as a disguise is illegal also adds to the importance of the information. The challenge being, both information pieces are hidden in the report documents. If not read by the same analyst and memorized correctly, chances are these two pieces of information will not be connected and the later analysis might go in the wrong direction. Thus, we recognize the key challenge from the aspect of information is the *incomplete information available to make a judgment.*

2.2.2 Additional Needs for Shared Artifacts and Common Ground

Collaboration in the sensemaking processes brings in different skills and perspectives, compensates for the different availability of experts, and can help mitigate many individual errors. Analysts from different organizations may have access to different documents; more readers can sift through larger amounts of data and generate more diverse perspectives to identify alternative patterns and hypotheses.

Collaborating on sensemaking tasks requires analysts to externalize their mental models and represent insights in an understandable way to each other. Research and tool development in collaborative sensemaking aims to support multiple analysts to explicitly work together. Analysts collaborate on sensemaking tasks by using visual analytics tools to forage for information; identify topics; and plan more in-depth analysis [41, 56, 62]. Metaphors like folders and bookmarks are used to organize fragments of information to create task-specific contexts [30]. Large displays where analysts can annotate, link, and spatially organize doc-

uments and named entities were proved to establish an efficient visual common ground that facilitates collaborative sensemaking [18]. Small groups tend to rely on shared interfaces and visual metaphors (such as node-link graphs) to co-create concept maps [42]. Such shared artifacts and metaphors are important for a group of analysts to collaborate synchronously on foraging for information, identifying topics, and planning more in-depth analysis [32, 62]. However, these and most other collaborative sensemaking projects focus mainly on certain sensemaking components [84, 184, 186] or assume the same analysts are involved in the entire session [39, 47, 173].

2.2.3 Hand-off Timing and Instruments for Asynchronous Collaboration

Synchronous collaboration can be constrained by expert availability and does not scale well with a bigger number of analysts. Asynchronous collaboration, however, introduces the challenge of handing off intermediate results between analysts. Mental models of analysts are usually considered as black boxes and the analysis processes are usually impromptu. This makes communication among analysts essentially difficult. Furthermore, prior work shows that the efficacy of hand-off heavily depends on the timing. In fact, hand-off in collaborative sensemaking is seldom successful unless it happens very early (transfer) or very late (referral) in the process [162].

Prior research has studied how to support the communication among experts and design instruments of hand-off to establish a shared understanding among analysts. Analysts examine the entities in the documents from different perspectives (e.g., categories, document contents) [14, 165] and data structures (concept map [41], bicluster [169]). Goyal et al. [75] found that visualizing data links is more effective than a notepad of annotations as an in-

termediate analysis artifact. However, schema and visual layout of the information [15, 168] is usually designed to best suit the mental models of previous analysts and are hard to understand without sufficient context and a detailed walk-through. Such hand-offs still rely on a shared understanding of the schema and visual layout of the information [10, 62, 163]. To help establish the shared understanding, Zhao et al. [198] developed Knowledge-Transfer Graphs that automatically capture, encode, and streamline analysts' interactions to support hand-off of partial findings during analysis.

Passing intermediate analysis among different analysts leads to loss of context and introduces inevitable overhead to make sense of the sensemaking by previous analysts. This extra cost can usually be mitigated by involving the same group of analysts who collaborate over a long period of time and would learn and adapt to each other's mental models. As a result, introducing new analysts to an existing team can be time consuming. The level of engagement required for collaborative sensemaking constraints the number of analysts that can be involved and therefore the scalability of our sensemaking processes.

2.2.4 Teammate Inaccuracy Blindness and Reluctance to Share Information.

In addition to the challenges for coordinating multiple analysts, collaborative sensemaking might lead to additional errors due to groupthink [91] that produces irrational or dysfunctional decision-making outcomes. Handing off intermediate results between analysts, meanwhile, introduces a new risk of sharing a premature focus on wrong suspects and can derail the overall investigation trajectory or amplify biases and error propagation among analysts.

Group biases might be caused by similar backgrounds of analysts or by individuals who mislead the group. Kang et al. coined the term "teammate inaccuracy blindness" [93] to describe

the phenomenon where previous work from a partner is assumed valid and useful without sufficient quality checks. Inaccurate information can be reused and premature focuses can be built upon by other analysts. On the other hand, analysts may fear their analysis is wrong and hesitate to exchange information and insights [86]. To address this issue, Goyal et al. [72] proposed a social translucence interface to raise analysts' self-awareness, shedding light on when and how distributed collaborative pairs share intermediate hypotheses to enhance the analysis quality. Social translucence interfaces balance the visibility and quality of analysis between distributed collaborative pairs, but it is unclear how well such approaches would scale to a large number of analysts.

Small group collaboration relies heavily on analysts spending enough time and attention understanding and building on previous work by others, and is challenged by the trade-offs between synchronization and translucence of the analysis progress by different collaborators. In this work, we explore how to support larger scale collaboration on sensemaking processes so that intermediate analysis results can be passed to subsequent analysts with a minimal hand-off learning curve and groupthink. We next consider existing crowdsourcing approaches in complex sensemaking work.

2.3 Crowdsourcing Complex Cognitive Tasks: Large-Scale Coordination

Some of the above-mentioned challenges for expert individuals and small groups can be alleviated in a crowdsourcing context. For example, the crowds can delve into significantly larger amounts of information with less fatigue and more diverse perspectives. The transient nature of crowd work creates a natural social translucence. It is also easier to require the use

of a certain artifact to promote sharing information with novice crowds. Researchers have found success in systems that leverage crowdsourcing for information synthesis guided by experts. Crowds have improved the quality of some components in sensemaking processes. For example, Wang et al. [182] use crowds to verify and remove duplicated database records identified by computers, and Soylent [11] used crowds to shorten and proofread text as part of a word processor. When solving more complex problems, the sensemaking processes become more iterative and integrated. Workflows and task designs play a more important role in crowd sensemaking. Below we review the expert intervention to prepare and guide crowd work, and the current solutions to coordinate crowd intelligence in complex sensemaking.

2.3.1 Expert Intervention to Prepare and Guide the Crowd Tasks

Most of the current research builds on the assumptions of perfect input data, predefined and clear guideline [24, 96] or specific goals [81, 177]. Providing crowds with ideal input and detailed task specifications can illustrate best-case scenario results. For example, crowds can plan itineraries when given detailed background information and bullet lists of traveling goals using the Mobi app [196]. In Cascade [35], the crowds were able to induce the hierarchical structure of categories, but researchers needed to manually break down the input data (Quora responses) into smaller text items. Crowds were also able to provide analysis and explanations on social data [187], but they were given nicely visualized and carefully selected charts, with hints and examples relevant to the tasks. Furthermore, crowd sensemaking also needs expert intervention to address the tension between the limited amount of data in micro tasks (local view) and the overall goal with the entire dataset (global view). Prior research has experimented with different amounts of expert guidance [104] and controlled versus free-form crowd collaboration [175]. To enable crowdsourced sensemaking in some open-ended problems such as brainstorming and designing, experts also need to provide real-time guid-

ance [25, 115] or heavy-duty centralized coordination [176]. Such assumptions and expert intervention illustrate the potential of crowdsourcing to solve complex problems, but at the same time confine the application of crowdsourcing from real-world problem-solving.

2.3.2 Crowd Sensemaking Workflows and Paradigms

Researchers have been striving to push the boundary of what is "crowdsoursable" [57], hoping to bring the benefits of the crowd's scalability, flexibility, and creativity to new domains. More complex sensemaking problems require both integrated processes and holistic views of the data and therefore more challenging for crowdsourcing.

Achieving complex goals with crowdsourcing relies heavily on organizing the roles and tasks for each worker (workflow and task design), and selecting the high quality work (quality control approaches) to produce the final outcome. Algorithmic approaches have been combined with crowdsourcing to address challenges such as information overload and lack of global context (e.g., [27, 140, 181]). For example, Mohanty et al. [140] combine crowdsourcing and computer vision tools to enhance the accuracy in face recognition. Alloy [27] optimizes the division of labor between crowds and machine learning algorithms to cluster high-dimensional, short text collections. The crowd–AI collaboration showed great success in information foraging and schematizing, but the crowd is still challenged by pooling distributed facts in a meaningful way to develop hypotheses in exploratory analysis [171].

For exploratory sensemaking problems, researchers have built expert knowledge into workflows and software to support crowdsourced analysis. Prior works have designed different tools and techniques for decomposing complex goals into microtasks, allowing people around the world to contribute meaningfully to different components of sensemaking processes. For example, Soylent [11] proposed a *Find-Fix-Verify* workflow to enable high-quality text edit-

ing. Little et al. [126] demonstrated *iterate-and-vote* workflows which produced higher accuracy in messy handwriting recognition. Crowd Synthesis [5] scaffolds expertise for novice crowds via a *classification-plus-context* approach, where crowds first re-represent the text data then iteratively elicit categories. CrowdForge introduces the *map-reduce* framework that enables crowds to "write articles, research purchase decisions, and conduct basic science journalism" [101]. For more complex problems, dynamic workflows with more than one deterministic path are designed and explored. Dai et al. formulated a decision-theoretic optimization function to automatically drive the iteration between tasks and subtasks in crowdsourcing workflows [48]. Turkit incorporates human computation as a function call with a crash-and-rerun programming model [125].

However, the novice crowd's lack of expertise and adaption on different tasks can cause crowd-specific challenges and errors. After the crowds complete micro-tasks, experts often need to curate the mixed-quality results [82] and solve the remaining problems. Chaining multiple crowdsourcing processes without the above-mentioned expert intervention could cause unexpected errors and problems. Novice facilitators [25] and crowds [113] are shown to be inadequate to adapt a given workflow and produce unsuccessful results as a consequence. Morris et al. explored the possibility of letting the crowd further breakdown given microtasks and "subcontract" the smaller tasks to their fellow crowd workers [142]. While crowds demonstrated potential to subcontract existing microtasks, it is unclear how subcontracting can be applied successfully in more complex problem-solving efforts with multiple interdependent steps. Crowdlines [67] found that exposing individual crowd workers to more information (high context) and less guidance (low structure) and using tournament-style workflows yields higher quality results, faster completion times, and higher completion rates in topic merging tasks. We take inspiration from Crowdlines' synthesis interfaces and parallelized, hierarchical workflows. But rather than synthesizing information into a sum-

mary of a given topic, we explore the possibility of leveraging crowds to solve mysteries, which requires discovering less obvious connections and developing hypotheses to uncover the hidden truth.

Parikh et al.'s [152] notion of human-debugging, originally applied to computer vision research, takes out each specific component in the computational system's pipeline and uses human subjects to transform the same input given to machine into the output. Drawing inspiration from this paradigm, and the CrowdForge framework for complex crowd tasks [102], **we modularize the components of the sensemaking loop [153] and decompose each step input into context slices so that distributed novice crowd workers can contribute meaningfully**.

2.4 Quality Control of Crowdsourced Analysis

The outcome of a crowdsourcing application greatly depends on crowd performance. While crowdsourcing is a powerful paradigm for accomplishing work at large-scale in a timely fashion, the advantage is accompanies by a significant challenge of quality control. The distributed work coming from a diverse pool of crowd workers with various experience and background contains an estimated one-third low-quality work [11].

2.4.1 Crowd Work Quality and Influencing Factors

The accuracy in crowd work depends on the task, context, and the baseline condition. Reported accuracy is often around 60% [6, 59, 98] and sometimes can be as good as above 90%. For example, crowds can create a global taxonomy of online question datasets with quality 80-90% of that of experts [35]. Open-ended tasks would typically yield 30% error

rates [11]. Willet et al. [187] proposed seven strategies to improve crowd performance and achieved 63% useful responses in the best results. CRICTO [40] reports that 73.98% of crowdsourced links in a sensemaking exercise were rated valid by authors. In some mixed-initiative systems [29, 50], no standalone crowd performance was reported. Many papers focus on indirect quality measures such as the number of responses [17], or subjective ratings of the tasks [143], rather than comparing crowd results to a gold standard. Crowds have demonstrated the promising capability of solving complex problems, but even the most successful systems cannot eliminate all the errors in the analysis. It remains unknown how imperfect parts of the crowd results may influence later analysis and the long-term outcome.

Various requester decisions beyond poor task design also influence crowd performance. Lack of workflow transparency [99] can decrease quality and volunteerism, and a higher number of perceived co-workers can induce social loafing [143]. In addition, US-only workers tend to outperform non-US workers [50, 187], and a qualification test [50] can improve task performance. Some studies recruited expert crowds [176] or volunteers from social media [17], who tend to have higher quality performance than those from paid platforms like Amazon Mechanical Turk (MTurk). In this work, we chose a low recruiting requirement (acceptance >90% without enforcing US-only crowds) to investigate errors made by a broad range of crowd workers.

2.4.2 Quality Control Challenges for Crowdsourced Sensemaking

Crowdsourcing as a paradigm applied to sensemaking problems is challenged by the tension between the microtask local view and the global goal, optimal decomposition of the process into hierarchical workflows and the data into task assignments, as well as management and quality control of a large-scale workforce.

Fragmented and distributed local data can cause irrelevant, missing, or incorrect judgements [35, 187]. Crowd analysis can also be focused on only a fraction of the given information due to unevenly distributed data. While devoted analysts have access to the entire data set to gain a rich understanding of global themes, paid crowd workers usually commit only a short period of attention and effort, and thus are only able to work with a small portion of the data. Distributing the data among local microtasks makes it difficult for workers to accomplish high-level synthesis tasks, such as identifying emergent global categories in the data. State-of-art solutions include increasing the amount of local data [130, 171], re-representing and condensing the raw data [6, 178], or iteratively revisiting the previous results [35].

Parallel analysis by many workers may lead to multiple interpretations of the same data. To avoid falling into an infinite loop of "categorizing the categorization", hybrid systems are introduced to recognize duplicates and conflicts in the analysis [82, 107] and reassign the edge cases to crowds [29] to consolidate the analysis.

The mechanisms of MTurk and similar platforms have been criticized for incentivizing low-quality work, such as random guesses [59]. Some crowd workers might not pay attention to the given input [98]. Other low-effort errors include unclear or speculative responses, inattention to details, or focusing on superficial facts [187]. Requesters can improve worker engagement with more formative instruction language [9], peer-evaluation [192] or even mutual reward dependency [88]. There are also visual analytics tools that support monitoring worker's task status and managing the overall workflow [103]. Reviewing other people's work can help workers improve their own results [110]. On social media platforms, people tend to engage in self-correcting rumors when encountering information conflicts [7].

High-quality results in previous works demonstrates the crowd's capabilities in solving a diverse variety of complex problems, as well as the efficacy of proposed methods. However, little research focuses on understanding the good-faith reasons why workers struggle, make

errors, and fail. **Our research addresses this gap and frames the findings within the broader sensemaking loop to make them relevant to many types of crowdsourced sensemaking and data analysis systems.**

2.4.3 Coordination Artifacts for Complex Work: Trade-offs between Predefined Guidance and Situated Adaptation

Crowdsourcing workflows decompose the complex goals into smaller, computationally sequenced *microtasks* that can be worked on by distributed, transient, novice workers asynchronously, as well as combining the completed microtasks into a final result [81, 125]. The microtasks are predefined with clear and specific instructions and action items, for example, multiple choices, image labeling, text translation, and so on. The decomposition and assignment of tasks can be done by experts [34], the system [102], other workers [114] or a combination of the above [81, 103].

However, the crowdsourced analysis from these projects were imperfect. For example, crowdsourced science articles generated by CrowdForge tended to overlook major empirical results. Consequently, requesters need additional support to manage and curate crowd results by visualizing crowd outputs and potential problems [103]. CrowdScape [158] combines information about worker behavior with worker outputs in interactive visualization to support the evaluation of complex crowd work. These systems can help to evaluate the crowd output in each step, but they are not able to trace error propagation or steer refinement of the analysis in pipelined crowd analyses.

Furthermore, in some complex sensemaking problems like mystery solving, the dynamic, ad-hoc nature of the analysis process may be too spontaneous to be captured and automated in any predefined workflow and task design [39]. Workflow-based approaches are challenged

by a lack of adaptation in exploratory analysis [156]. The connections among facts and evidence hidden in the data won't be discovered until the relevant information is analyzed together. At the early stage of the investigation, information is extracted in an unplanned order thus it is almost impossible to connect the dots on the fly, especially in a crowdsourcing setting. To this end, prior work applied human evaluation and feedback to help crowds adapt the workflow in situ in response to the unplanned contingencies. Real-time systems maintain a pool of stand-by workers with the retainer model and can support crowd-powered workflows by dynamically adapting the tasks [13, 16]. Requesters can also collaboratively design workflows with workers [114] or provide timely, task-specific feedback [53]. Building on the prior research efforts, **we address this challenge from a "top-down" perspective — examining pipelined crowd analyses, tracing error propagation, and steering crowdsourced refinement of the analysis with the knowledge and insights newly learned from the dataset.**

2.4.4 Techniques and Best Practices for Better Crowd Outcome

Researchers have investigated both algorithmic and manual assessment approaches to prevent poor performance before the tasks, detect and/or correct the low quality work during and after the tasks.

Pre-task approaches: prepare the tasks and select the crowds. When using crowdsourcing, requesters usually need to decompose a real-world complex problem and select well-defined sub-components. For example, when building a machine learning model, researchers recruit crowd workers to develop training data by labelling the objects in images [147]; when evaluating a search engine, researchers recruit crowd workers to judge the relevance of a given search result [2].

Researchers also need mechanisms to ensure and enhance the quality of crowd work in the microtasks. Guidelines and best practices for microtask design have been proposed increasingly in recent research. Common practices include selecting workers by reputation, screening workers with qualification tests, advocating the intrinsic motivation [90], providing ideal input [33] and task-specific guidelines [97, 195], or tweaking the outcome measures for extrinsic motivation such as monetary reward [138]. Good task designs should also make it no easier to game the task than to contribute as instructed [100]. Writing clear instructions [191] and providing examples [65] help communicate requester intention to crowd workers. Gated instructions [128] that provide interactive tutorials and screening questions can test the worker's understanding of the task. Grady et al. [76] found that wording and terminology are also important in task design. Formal language was shown to improve participant attention in both paid and volunteer crowds [8]. Researchers also spent efforts on building expert knowledge into exquisite workflow and task design. In addition to developing human computation algorithms that split larger tasks into small, fault-tolerant sub-tasks [182], it was shown that incorporating randomness in the task workflow and assignment can also be helpful for improving the quality of crowd work [179]. Apriori algorithms can be effective in preventing common or known mistakes, but it is difficult to fully predict and control all human errors.

However, designing clear and actionable tasks is nontrivial and usually requires several iterations with implicit and explicit crowd feedback [111]. Turkomatic [112] introduced a meta-workflow that allows the crowd to "collaboratively design and execute workflows in conjunction with a requester". Building on that work, Sprout [20] enables requesters to iteratively refine task design with worker feedback. Fantasktic [79] allows novice requesters to create more systematic task instructions via a wizard interface that display a preview of the microtask interface. WingIt [136] implements methods to enable workers to cope with

unclear or ambiguous instructions and produce high-quality results with minimal reliance on the requester. In this work, we build on ideas from Fantasktic and WingIt to scaffold microtask creation with novice auditors, as well as evaluating the quality of microtasks. We embed the microtask creation process as part of crowd auditing and support the corresponding context switch from identifying problems with the existing crowd analysis to assigning microtasks.

Within-task approaches: real-time guidance and feedback. While crowds work on the microtasks, researchers can also enhance their quality of work by providing rubrics for self-assessment, or timely feedback while workers work on a subsequent task [53]. Requesters also need to monitor and manage the different pieces of the work assigned to different crowds. CrowdWeaver [103] uses directed graph visualizations to show the organization of crowd tasks, allowing requesters to better understand and manage crowd workflows. Other approaches focus on real-time collaboration between requesters and workers. Through directly and continuously communicating with workers via live audio streaming [4, 115, 118], the requester can monitor and guide crowds in their work, providing feedback for the crowds to refine their results simultaneously. This broadens the applications of crowdsourcing to more complicated and open-ended problems, but requires heavy-duty involvement on the requester side, and does not scale well with a high volume of data and a large number of workers.

Post-task approaches: identify and drop the bad results. Aggregating and managing the crowd results also play an important role in controlling the quality of the crowd outcome. For example, researchers can select high-quality work with validated gold standard data [22, 55], or when gold standard data is not available, aggregate crowd results by using worker agreement (majority vote), peer evaluation and review to control or correct worker

outputs [126, 200]. In addition to majority vote [21, 49] and iterative improvement [127], requesters can employ defensive task design to inform the post-hoc evaluation of task quality in an exploratory analysis. Embedding gold-standard test questions in the workflow to test worker diligence [19, 148], manipulation checks to test worker attentiveness [149], and behavioral traces [159] can provide performance indicators and help filter out poor-quality results. Effective as these approaches are, they still cannot eliminate all the low quality work and might suffer from majority effects when most of the crowd workers make the same mistakes, or even break down when there are no answers in common. Another line of research by Rzeszotarski et. al captures implicit behavior traces of crowd workers and build predictive models to infer the task performance [160]. Workers submitting the results unnaturally fast could be of poorer quality than those who spend a more reasonable amount of time. Evidently, such approaches require an expert who understand both the problem and crowdsourcing to set a threshold of what behaviors can be entitled as "reasonable" and what would be "unnatural".

Algorithmic approaches can be applied at a large scale promptly, but are constricted with deterministic or constrained tasks [53, 102, 103]. Human evaluation can compensate algorithms to evaluate subjective and situated judgment in complex problems but does not scale well [118]. Integrated methods that support large-scale human evaluation through predictive models and interactive visualization of worker behaviors have seen success in increasing the efficiency of identifying good work [157]. Nonetheless, the current quality control approaches primarily focus on *obtaining good data* from the crowd workers. **We explore and take a more in-depth look at the where and why the crowds are challenged in solving complex problems and discuss the design opportunity for refining and building on previous crowd analysis.**

2.5 Providing Feedback in Sensemaking

Sensemaking offers great potential to understand the meaning and patterns contained within large quantities of unstructured, noisy source materials. It involves continuous and iterative development of a mental model, or a conceptualization, from the schema/frame that best fits the evidence/data [109, 154].

Making mistakes is inevitable in sensemaking. Individual analysts can sometimes miss or misinterpret important information due to inaccurate memories, cognitive fatigue, and perception constraints [37, 86]. Collaboration in sensemaking helps mitigate many individual errors but does not eliminate all possible errors. Collaborative sensemaking also incurs new challenges, such as designing effective communication artifacts [18, 31, 42, 61], hand-off overhead [15, 74, 199], groupthink [145] and teammate inaccuracy blindness [92]. In crowdsourced sensemaking, crowd workers are limited by local views of the data to adequately pool distributed facts and therefore subject to biased and unstable analysis [171]. Chaining multiple crowdsourcing processes without the expert facilitation usually results in error propagation that compounds different mistakes in different steps [123].

Feedback is, therefore, an equally important discovery pathway in sensemaking processes to correct different kinds of errors [108]. The data-frame model of sensemaking [109] describes feedback as "discovering inadequacies of initial account, comparison of alternative accounts, reframing the initial account and replacing it with another". The sensemaking loop model [154] involves top-down processes that test theories against the data to validate the analysis outcome in each sub-process. Feedback is also referred to as confirmatory analysis [174], or retrospective sensemaking [85, 124], in different domains of sensemaking.

The most common sources of feedback in sensemaking processes are clients [154] and peers [161]. Peer feedback generally involves analysts who share the same problem-solving goals inspect-

ing one another's work [37]. Self-assessment has also proven useful, achieving comparable results to external sources of feedback [52]. Finally, novice crowds have demonstrated the potential to identify a wide range of problems and provide high-quality feedback in design domains [131]. In this work, we draw on the prior research and further explore who and how to provide feedback for a pipeline of crowd analysis in exploratory sensemaking.

Chapter 3

Context Slices and Crowdsourced Relationship Graph Building

In this chapter, we explore the use of non-expert crowds in the sensemaking loop of expert analysts, by restructuring the dataset into context slices for each crowd worker, such that they can do independent, in-depth analysis. We reified this technique in a web application, *Connect the Dots*, that visualizes the dataset and enables crowd-created connections. We experimented different slicing methods and their impact on the quality of crowd analysis. We also contribute ways of aggregating crowd-generated connections that support strategic information retrieval and schematizing by expert analysts in large-scale text data analysis.

3.1 Connect the Dots: System Description

To help crowd workers analyze documents and make connections between entities, we designed a web application called Connect the Dots (Figure 3.1). We built the application using a Python/Django back-end with a Postgres database, and Bootstrap and D3.js for the front-end. There are two main features in the web application to facilitate each crowd worker's analysis process within a given context slice: 1) the Document View and 2) the Connection Workspace.

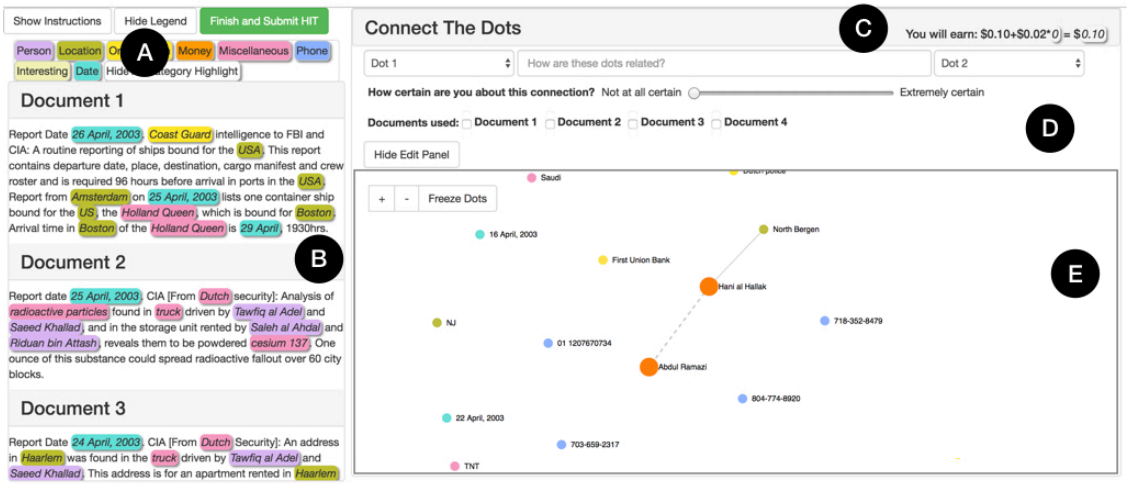

Figure 3.1: The Connect the Dots web application interface

3.1.1 Document View

The left side of the interface lists all the documents in the context slice (Figure 3.1.B). The entities are automatically extracted from the documents with a named entity recognition algorithm, and highlighted in different colors by categories: person, location, organization, money, phone number, date and miscellaneous. A legend is provided to describe the category names and colors (Figure 3.1.A). Workers can click a category name to show or hide all entities of that category in the document(s). They can also toggle all the category highlights on or off, and the entire legend can be hidden to save display space.

3.1.2 Connection Workspace

The Connection Workspace is composed of the visualization panel and the edit panel, both on the right side of the interface.

The visualization panel (Figure 3.1.E) displays the entities in documents as nodes ("dots"), colored based on their categories and labeled with the entity names. When the user selects two unconnected nodes, a dashed line appears to suggest a potential link. Once the link is created between the two corresponding entities, the line becomes solid black. Selected nodes and existing edges are highlighted in a thicker orange stroke. Only the most recently selected two nodes are highlighted. Selecting an edge will automatically select the two nodes it connects. The worker can zoom and pan the visualization via the buttons on the upper left of the panel. By default, the visualization uses a force-directed layout to minimize overlaps and intersections, but the user can click the freeze/unfreeze button to control the graph movement and manipulate node positions via drag-and-drop.

The edit panel (Figure 3.1.D) is an input form where users can create and describe node connections. Four types of information are required for each connection: 1) the names of the two nodes to be connected, 2) a brief description of their relationship, 3) the user's certainty about the connection, expressed by moving a slider, and 4) checkboxes to indicate which documents provide evidence supporting the connection. When the user selects two entities with no connection between them, a "Create Connection" button appears. If a connection already exists between the nodes, then "Update Connection" and "Delete Connection" buttons appear instead. Users can hide the edit panel to save visualization space.

Users can select the nodes to connect in any of three ways: 1) choosing from alphabetized dropdown menus in the edit panel, 2) clicking on entities in the documents, and/or 3) clicking on the nodes in the visualization.

Finally, the user's number of connections made, and the corresponding payment earned, are updated on the upper right every time the user creates or deletes a connection (Figure 3.1.C).

In the next section, we describe an experiment to evaluate the utility of the Connect the Dots system and the context slices approach.

3.2 Study Design

The goal of this study was to answer the following research questions:

RQ 1.1: What types of connections does the crowd create?

RQ 1.2: How do different slicing methods influence the crowd results?

RQ 1.3: When using context slices, how well can crowds find the connections needed for the solution?

RQ 1.4: When using context slices, how can we distinguish or prioritize the most important entities?

3.2.1 Dataset

We use a subset of the Sign of the Crescent dataset [89] developed for the purpose of training professional intelligence analysts. The original dataset consists of 41 report documents regarding three fictional terrorist attacks. Each document contains a single prose paragraph ranging from 33 to 210 words.

In this study, we focus on solving one of the three plots in this dataset. The relevant information for this plot is distributed across 10 of the documents.

Creating context slices. From our pilot studies, we found that a slice size of one or two documents usually takes 15 to 30 minutes for one crowd worker to finish, depending on the number of entities and other words in the documents. This is considered a reasonable amount of work as a micro-task [66]. Therefore, we generated 55 possible context slices: 45 different combinations of double-document slices and 10 single-document slices. This covers three types of slicing methods: single-document slice, double-document slices with overlapping entities, and double-document slices without overlapping entities.

Gold standard connections. To evaluate crowd-generated connections, we created a set of gold standard connections to compare. The Crescent dataset provides, as a kind of answer key, a list of important information pieces necessary to uncover the hidden plot, as well as a hierarchical graph presentation that describes the deduction process and higher level hypotheses derived from the important information pieces. Since our focus is on extracting important connections, rather than uncovering the entire plot, we needed to adapt these materials for our purposes. In order to be objective and adhere to the given solution, an author of this chapter generated a set of gold standard connections by making connections between entities that appear in the same sentence in the provided solution materials. This approach yielded 177 gold standard connections. Our assumption is that the more crowd-generated connections match the gold standard connections, the better the crowds are performing. The same author also generated a gold standard edge label for each connection, but because this process was more subjective, we evaluate it differently, as described in detail below.

Algorithmic baseline. In addition to the gold standard connections derived from the answer key, we also generated an algorithmic baseline of entity connections based on document co-occurrence. This approach yielded 790 connections for a baseline.

3.2.2 Participants

We recruited crowd workers from Amazon Mechanical Turk (AMT), restricted to US-only workers with an acceptance rate greater than 90%. In total, we recruited 275 crowd workers and randomly assigned five workers to each context slice.

Each worker was unique and assigned to only one HIT (Human Intelligence Task) on AMT, to mitigate learning effects or collusion. A crowd worker who returned (quit) an accepted HIT without submitting it was not allowed to resume the unfinished work or take a new HIT.

Workers were required to make a minimum number of connections based on the number of entities extracted in the given context slice. We found in pilot studies that an explicit minimum number of expected connections should be specified in task instructions. We compute this minimum requirement based on the number of entities in each context slice (e.g. a context slice with N unique entities are expected to have at least N/2 connections). To motivate productivity, we paid workers $0.02 on top of the base payment $0.10, for every extra connection they make beyond the minimum requirement.

3.2.3 Procedure

After accepting the HIT, each worker was randomly assigned to a context slice. Each task starts by showing the worker an online IRB consent form. If the worker accepts, she will

see a modal dialog box with HIT instructions. The instructions explain the background and documents ("a few pieces of evidence from a fake terrorist plot"), the task ("make connections based on the information"), how to use the interface (a numbered list that explains the steps of selecting entities and inputting results), the minimum number of connections required, and the bonus policy. Workers can close the instructions, which will reveal the task interface, and can click the "Instructions" button to reopen them at any time. The "Finish and Submit HIT" button stays disabled until the minimum number of connections is made. Once workers have connected enough pairs of entities, they can click the "Finish and Submit HIT" button and voluntarily provide feedback.

3.2.4 Data Collection

For each worker who accepted the HIT, we collected their basic AMT credentials (worker id and assignment id) to identify unique workers, the id of the context slice they were assigned to, the time when they accepted and submitted or returned the HIT, the working status (if they accepted, returned or submitted the HIT and the amount of bonus they were granted for submitted work), the connections made by them, and their feedback, if any. For each connection, along with the entity pair and annotations (including relationship descriptions, evidence documents and level of certainty), we recorded the timestamp when a connection is made, the worker who created it, and the context slice from which it was created.

3.3 Results

We first inspect a sample subset of crowd-generated connections to gain a basic understanding of resulting crowd analysis and set the ground for further in-depth evaluation. Then

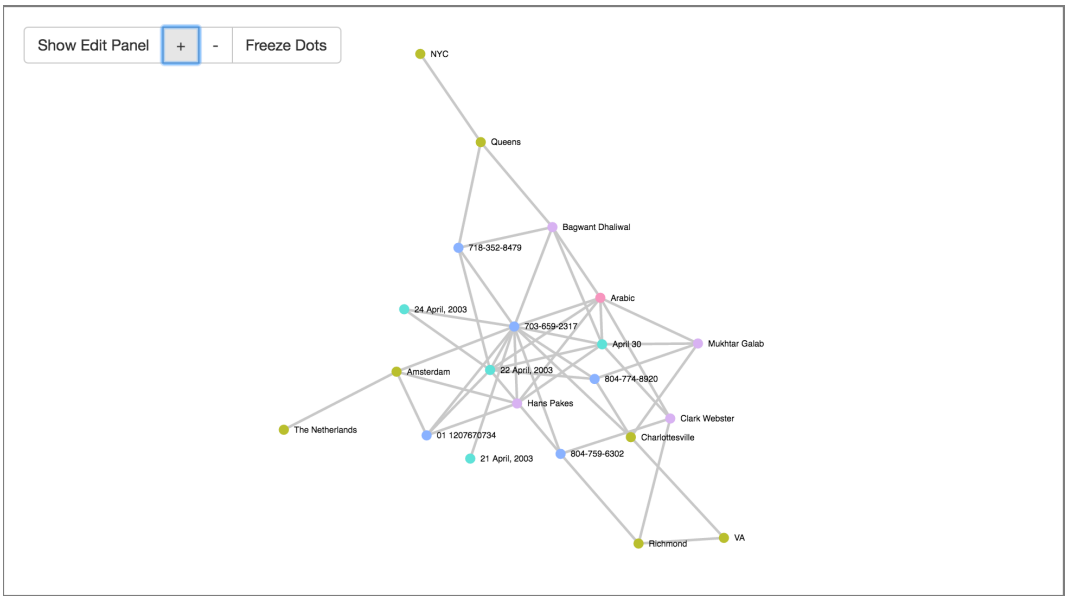

Figure 3.2: Example subgraph of connections made by five crowd workers for one context slice.

we conducted qualitative analysis to compare overall statistics of each slicing methods and the precision recall value against gold standard connections. After that, we run clustering algorithm on crowd-generated descriptions for entity pairs, and applied one common strategy to retrieve and schematize information using crowd-generated connections. We also use algorithm-generated co-occurrence connections (790) as a baseline.

3.3.1 RQ 1.1: Types of Connections

Since our bonus policy encouraged the crowds to create extra connections, the experiment results in a large number of connections from the crowds. In order to better understand the resulting analysis, we randomly sampled 727 of the 5992 crowd-generated connections and inspect them in detail. Also considering the nature of machine-recognized entities [60], we identified three types of connections:

T 1: Contextual Connections. This type of connection represents the semantic relationship

given only in the documents. The entities cannot be connected without the information given in the contexts. For example, a person, Hans Pakes, uses the phone number 703-659-2317.

T 2: Common-sense Connections. This type of connection represents common sense or ground truth related to the entities being connected. For example, Queens is a borough in NYC.

T 3: Collateral Connections. This type of connection represents meta information of the documents and entities and do not convey human intelligence. Such connections can be generated as well or better by algorithms, yet still benefit the analysis. For example, April 30, 2003 and April 25, 2003 are both dates.

The contextual connections (T1) are the most important information that leads to solving the hidden plot. This category of connections can be further classified by different level of difficulty: whether the information is explicitly stated in a document (level 1), or it requires several level 1 connections from multiple documents combined to make the connection (level 2), or it requires the analyst to take a risk and make a hypothesis (level 3).

The common-sense connections exist for two reasons. One is that it is challenging to customize an NER algorithm that chooses the perfect granularity of entities for a given analytical purpose. Another reason is that realistic documents present entity information in inconsistent ways. For example, "… give her address as: [1631 Webster Ave.] [The Bronx.] [NYC]." and "obtained a [social security card] and a [New York] State [driver's license] in [Queens]" are two parts of sentences from two documents, with machine recognized entities wrapped in brackets. For the first sentence, it is better in this case if the three entities are merged into one address, yet [The Bronx.] and [NYC] might reappear individually in other sentences. For the second, we cannot penalize workers if they connect [Queens] with [New York], as

they are indeed related. Furthermore, worker who sees both sentences might connect [New York] and [NYC] as well, which do not provide contextual information but will appear as multi-document connections. Such being the case, common-sense connections are not trivial to avoid by simple merging the entities beforehand.

The collateral connections are metadata about the documents that do not contribute to the sensemaking process. These could be filtered out by an algorithm, or prevented with interface feedback. For example, when a crowd worker tries to make a collateral connection, she might describe the relationship as "both [category name]". If the system (designer) learns the patterns of such connections, it can issue a warning to eliminate such results. However, accurately detecting T3 connections may be time-consuming to implement and comes with risks of false negatives.

Therefore, we did not clean the crowd-generated connections to remove T2- and T3-type of connections in our following analysis. Instead, we augment this typology with a qualitative examination of the missing gold standard connections, and some sample extra connections made by the crowds.

3.3.2 RQ 1.2: Comparing Slicing Methods

The 275 crowd workers created a total of 5992 connections from the 55 context slices in total (mean=23.5, SD=14.3). This includes connections between the same pair of entities by different workers. In total, 622 pairs of entities were connected by crowd workers. The average time spent on each HIT was approximately 20 minutes (min=8.5, max=37, SD=6.3).

Specifically, for single-document slices (10 slices), the crowds created 671 connections (mean=13, SD=11.6) between 304 pairs of entities. For double-document slices with overlapping entities

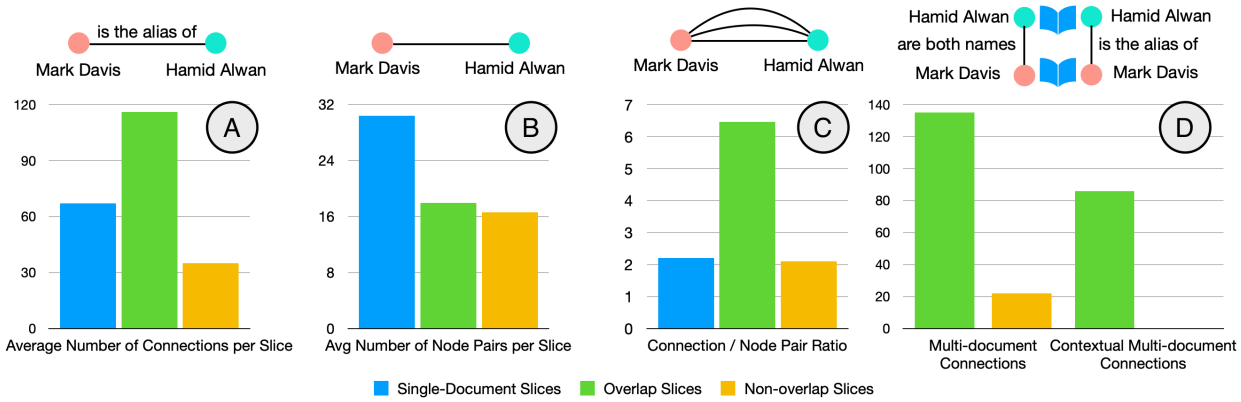

Figure 3.3: Average number of connections per slice, and multi-document node pairs in different slicing methods.

(26 slices), there were 3019 connections (mean=28, SD=16.75) between 467 pairs entities. For double-document slices without overlapping entities, we take an example of 5 slices that cover all 10 documents, there are 666 connections (mean=19.4, SD=3.6) between 316 pair of entities (Figure 3.3).

As in double-document slices with overlapping entities, some of the documents were assigned to more than one group of crowd workers. We computed the average number of connections in each slice to normalize this difference. We can see in Figure 3.3.B that overlapping documents lead to more than double the number of connections, while non-overlapping slices lead to less than double the number of connections, even as the number of documents is doubled. This indicates that increasing the amount of work without bringing in shared contexts will not increase, and may even hinder, crowd productivity.

Double-document slices led to connections between entities pairs marked with more than one evidence documents (Figure 3.3. D). Since the same information can appear in different documents more than once, we also computed "non-trivial" multi-document connections, by counting only the connections where the two entities came from separate documents. Although this cannot fully guarantee the importance of the connection, it eliminates all

single-document T1 (contextual) type connections that re-appear in multiple documents. The double-document slices without overlapping entities appear to have 20 non-trivial connections, but a closer examination reveals that all 20 connections are common-sense information. In contrast, the 86 non-trivial connections in overlap double-document slices contains information that requires both documents. For example, Abdul Ramazi—April 30: 'Reported will be in office at this time'. This connection requires reading two documents, one containing the person's name and phone number, the other containing the phone number and the message (T1 level 2 connections). We also observe several T1 level 3 hypotheses, e.g.: Hamid Alwan—Hani al Hallak: 'al Hallak may have supplied explosives to Alwan'. This is actually one of the hypotheses given in the solution.

3.3.3 RQ 1.3: Finding Key Connections

To understand how well the crowds can retrieve the connections needed for experts to uncover the plot, we compute the precision and recall values using entity pairs connected by crowd workers against the set of gold standard connections G (177). Given a set of crowd-generated entity pairs C, the overlapped entity pairs O=C∩G. The precision value is then computed as $P = (|O|)/(|C|)$ and the recall value is $R = (|O|)/(|G|)$.

For each slice, we used the number of workers that connected a certain pair of entities as a "majority vote" (1-5) threshold to decide whether to count this entity pair in the result or not. For example, if the threshold was 3, then only entity pairs that were connected by 3 or more out of the 5 workers working on this slice were considered. The results from each slice were then aggregated to produce a set of crowd-generated entity pairs for each threshold. Let the set of connections of the ith context slice with threshold t be C_i^t ; the set of combined connections given a threshold t is $C^t = \cup_i C_i^t$. Precision, recall, and f-measure

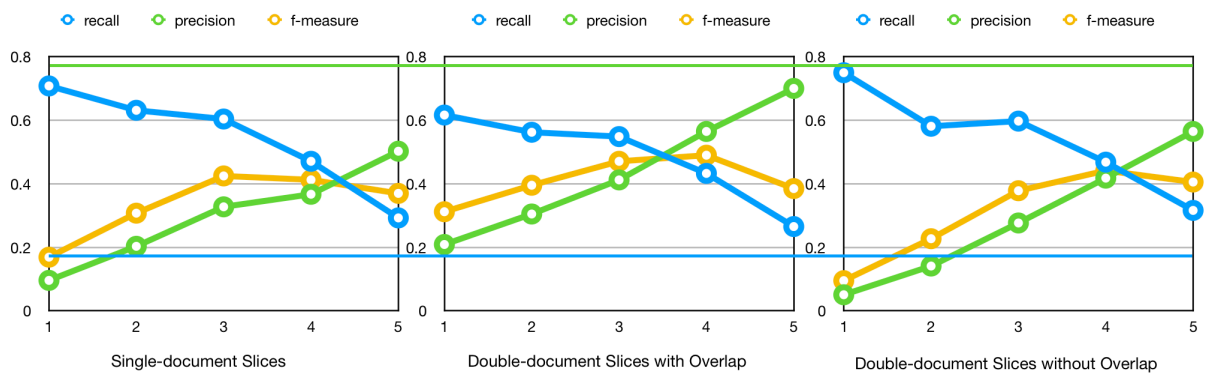

Figure 3.4: Precision, recall, and f-measure values for varying worker vote thresholds.

(harmonic mean of precision and recall) for each slicing method using combined connections in each threshold are shown in Figure 3.4. Our algorithmic baseline generated by document co-occurrence gave a precision value of 0.17 and a recall value of 0.77 (the two horizontal lines in Figure 3.4).

The overall precision-recall values are similar between single-document slices and double-document slices without overlapping entities, reaching optimal f-measure at threshold = 4. Double-document slices with overlapping entities produce the best crowd results, with a maximum f-measure of 0.50 with a threshold of 4 workers.

With a threshold of less than 3, the f-measure of non-overlap slices is less than 0.4. This indicates that slices that contains overlapping contexts will lead to more stable quality from crowd-generated results. Double-document slices with overlapping entities also require one less crowd worker (3 vs. 4) to achieve a better f-measure than the other slicing methods.

Even with a threshold of 5, single-document slices and double-document slices without overlapping entities can only recall around half of the gold standard connections, while double-document slices with overlapping entities outperform by 50% to achieve a recall value of 0.75. This is close to co-occurrence connections (0.77) but with 60% the number of node

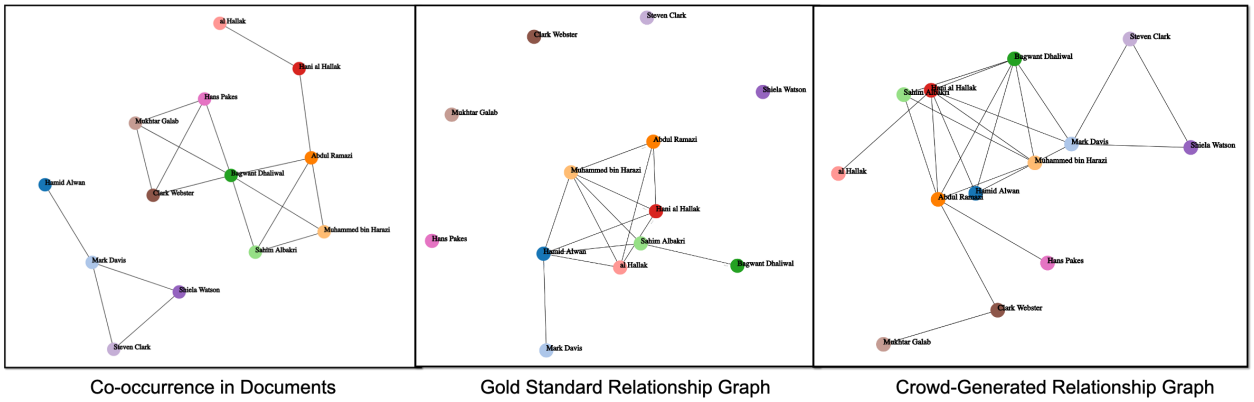

Figure 3.5: Relationship graphs of person names by document co-occurrence, gold standard, and crowd workers.

pairs (467 vs. 790) being connected.

We also examined the gold standard connections that the crowd were not able to create. In the 40 gold standard connections not created by any crowd worker, 23 are T1 level 3 connections generated from expert hypotheses, or T1 level 2 connections that require more than two documents to connect. The remaining 17 connections are synonymous with connections that were created by crowd workers. For example, some gold standard connections use the surname entity al Hallak, but the crowds used the full name Hani al Hallk entity to connect to the same nodes.

The existence of T2 connections results in a lot of noise in crowd generated connections, even though the crowd outperforms the co-occurrence baseline. To address this, we apply a common strategy [39] for schematizing information in intelligence analysis: investigating and aggregating relationships between person names. We visualize both the crowd-generated and gold standard connections for person names, to evaluate the quality of crowd-generated connections.

We found that crowd workers successfully connected and correctly described all pairs of person names whose relationship can be discovered using two documents. Figure 3.5 visualizes

crowd-generated connections (left), gold standard connections (middle) and document co-occurrence (right) regarding person names. All of the circled person names in gold standard graph are connected to more than two other person names in the crowd-generated graph, whereas no similar patterns were found in the baseline co-occurrence graph. This indicates that the crowd-generated graph of person names can accurately identity top suspects and get experts started on further investigation.

It is also possible to learn the relationship between people by reading the most frequent crowd-generated labels for that connection. For example, the connection "Bagwant Dhaliwal—Sahim Albakri" is most frequently described with the words: 'indian', 'alias', 'used', 'name', 'passport'. It can be inferred that these two names are used by the same person and it is even (correctly) suggests the fake name is used in an Indian passport. With a quick review of the original descriptions written by crowd workers, expert analysts can easily retrieve the relationship information about these two names.

Based on these observations, we explored using a clustering algorithm to computationally aggregate useful connection descriptions. We ran K-Means algorithm based on tf-idf similarity between edge labels to cluster them. A quick ranking of description labels for each pair of entities reveals that there are many identical descriptions written by different crowd workers. In addition, non-identical descriptions are often very similar, with many repeated key words (e.g. "city in state" vs. "city is in this state"). Considering that common stop words are useful to convey information in our case ("is in, are from, etc."), we only used four stop words: "the", "a", "this", "that". For each description, we first removed the words in the two entities it connects then the four stop words. If there were still words left, the remaining words in the description were then tokenized and stemmed before computing tf-idf similarity.

We tested cluster numbers of 3 and 10 for a K-Means algorithm to understand the number of

relationship categories the crowd generated for each node pair. Both numbers yield highly similar top words in each cluster. After removing the duplicate top words, the overall centroid words in all clusters were less than 10. In almost all cases, the combined top words provided valuable semantic information to convey the relationship between the entity pair. For example, the connection between two person anmes Hamid Alwan and Mark Davis has the top centroid words: ['person', 'as', 'name', 'same', 'identified']. Example of crowd-generated connection descriptions are "name used by" and "same person". Thus, despite the minor differences in description labels, the keywords used to portray the relationship between connected entities are usually similar, and can be aggregated using representative centroid key words. This preserves the semantic meaning of the description and can be understood without reading either the crowd's description labels or the original documents (those two names in the example refer to the same person identity).

3.3.4 RQ 1.4: Prioritizing Important Entities

Putting all crowd-generated entity pairs together, we can rank all possible entity pairs by the number of workers connecting them (i.e., votes). In Figure 3.6, the x-axis spans all 8128 possible combinations of entity pairs (128 entities) that appeared in the 10 documents. The y-axis shows the number of workers who connected a certain pair of nodes. The gold region (left) of the bar plot shows the node pairs from gold standard connections, while the blue region (right) shows the remaining ones. The long tail of the plot (entity pairs that were not connected by any crowd workers) is truncated to show a more detailed view.

The plot shows that the gold standard part of crowd-generated connections has a different distribution than that of the non-gold standard part. There is a plateau in the golden part while the blue part is right skewed. A t-test shows that pairs in the gold standard set have

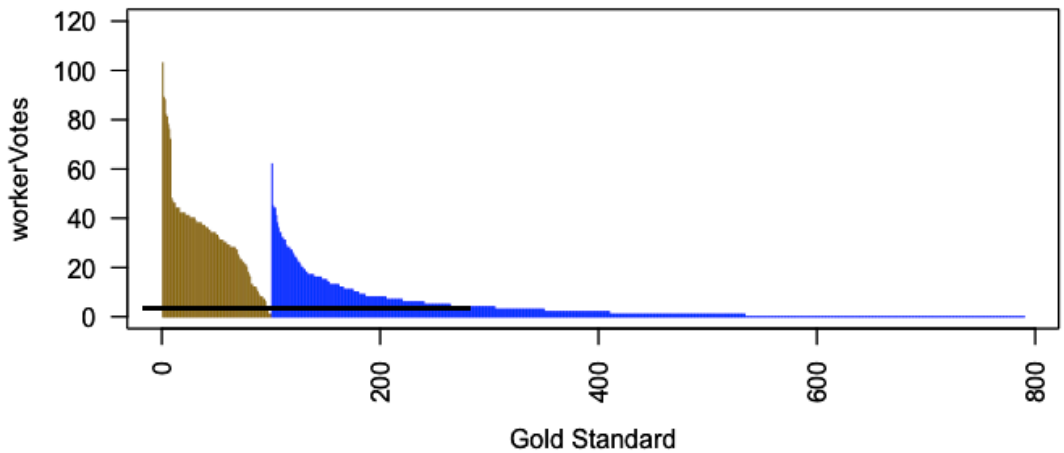

Figure 3.6: Ranked entity pairs by the number of workers connecting them. Gold lines are entity pairs from the gold standard. Blue lines are other possible entity pairs.

a significantly higher vote count than those in the non-gold standard set ($t = 19.257$, $df = 2335$, $p < 0.001$). Thus, the number of votes can be a useful guide for experts to exploit these connections, by focusing on the highly voted pairs.

We also ranked entities by their degrees, i.e., the number of connections the entity has in a graph, for both the crowd-generated graph and the gold standard graph. There are 85 total entities in the 10 documents. The entity with the most links is ranked No. 1 and the entity with the fewest (if not 0) links is ranked No. 85. Figure 3.7 shows entity ranks for connections in the crowd-generated graph (x-axis) and ranks for gold standard connections from the solution graph (y-axis). The points are colored from red to blue, where points of smaller rank (higher degree) are more red.

The rankings computed in both graphs are very similar ($r = 0.89$). Thus, the connectivity of entities in the crowd generated graph can be a useful guide to help experts locate important entities.

Figure 3.7: Rank entities by their degrees in the graph.

3.4 Discussion

We conducted experiments with 275 crowd workers by applying three different context slicing methods on a 10-document dataset about a fictional terrorist attack plot. We identified three types of connections that the crowds generate, and characterized the challenges of avoiding or excluding less useful types. By comparing quantitative analysis results among different slicing methods, we found that double-document slices with overlapping entities provide shared contexts between documents and outperform other slicing methods. Single document slices are efficient in terms of collecting contextual information pieces, but lack the ability to generate insightful non-trivial connections between documents. Using more than one document without overlapping entities will hinder the quality of work and is not recommended. Qualitative analysis of the crowd-generated connection descriptions (edge labels) shows that crowd can successfully structure the source documents in ways that could help experts strategically retrieve and schematize important information. In addition, the crowd-

generated descriptions appear to converge well and provide accurate and understandable labels for each connection.

3.4.1 Context Slicing with Overlapping Entities

We compared analysis results among slicing methods to understand their impact on the quality of crowd analysis. Almost every analysis between simple document slices and double-document slices without overlapping entities has similar patterns, which indicates that naively doubling the amount of work will not improve quality nor efficiency. On the other hand, double-document slices with overlapping entities shows the potential to intrigue more insightful high-quality connections using both documents. Crowd workers do not naively copy the text from documents to given answer boxes. They read, think, and make richer connections when given more context.

3.4.2 Coverage of Gold Standard with Thresholds

In connections from different slicing methods, the crowd was able to find about 75%, or 133 of 177, gold standard connections. We inspected the 40 connections missed by crowd workers. Twenty-three were made from more than two documents or expert hypotheses, which crowd workers were not given enough context in this study to connect. The remaining missed ones are synonymous to the gold standard, but linked slightly different entities. This indicates that crowd workers can reliably extract meaningful and useful information from large numbers of textual documents. However, because some gold standard connections required connections across three or more documents, it may be worth testing larger context slices, or going a step further in the sensemaking loop to ask crowd workers schematize their connections. This increased responsibility for crowd intelligence has demonstrated

potential, as we already observed insightful hypotheses created by workers, even using just two documents.

Since our web application paid a bonus to crowd workers for extra connections to motivate them to create more connections, the number of crowd-generated connections far exceeds the number needed by expert analysts. This helps explain the high recall value and low precision value when we set the majority vote threshold to 1, i.e., considering every connection made by each crowd worker. However, if we threshold the crowd-generated connections and consider an entity pair as valid only if it is connected by two or more crowd workers, we achieve a higher precision value while maintaining a reasonable recall. Our results also indicate that a threshold of three crowd workers is probably sufficient for such tasks, if double-document slices with overlapping entities are used.

Our study showed that crowd workers did not connect every possible pair of entities, even when they are paid a bonus for making more connections. We identified a handful of cases where crowd workers were gaming the task, but we can reduce the noise (e.g., false or meaningless connections) by ranking the node pairs by the number of workers who connected them. Using this approach, we reduced the number of possible connections without losing potential clues. More importantly, the results show that the more crowd workers make connections to an entity pair, the more likely an expert will need this connection in an analysis process leading to the solution.

3.4.3 Sources of Strategic Information Retrieval

In Figure 3.5, we presented an example scenario of how expert analysts might benefit from using crowd generated connections. In this example, the crowd-generated connections are further schematized with intelligence analysts' expertise, with a specific query about rela-

tionships between person names. Such strategies may be triggered by an expert's previous experience or access to additional documents beyond a worker's small context slice. Taking a meaningful subset of the large graph allows analysts to efficiently retrieve the desired types of information without having to go through large numbers of source documents. Likewise, the crowd can help experts avoid missing some seemingly irrelevant but actually important documents (e.g., documents that describe clues about the suspect's alias, but might not mention the suspect's real name at all).

We recognize the crowd-generated connections face the challenges of imperfect machine-recognized named entities and diverse terminology used in description labels. With description labels, it is likely impractical to enforce predefined language given the unpredictability of topics in the domain of intelligence analysis. However, a simple clustering algorithm can provide good insight into the convergence of relationship categories between node pairs while preserving semantic meanings. In the final graph of entities for experts to use, the edge labels are either cluster centroid top words or, if the number of descriptions are fewer than the cluster number, a list of raw descriptions by crowd workers.

Ranking nodes according to their degrees in the graph of crowd-generated connections yielded similar results to those from the gold standard connections. The crowd shows strong potential in finding important entities and make connections with them. This ranking can serve as a starting point in expert analysts' sensemaking process to help guide and refine their search of the solution space and prioritize entities within the same contexts. Such starting points are more concrete and contextual than prior list of types of terrorist attacks, since the list is never exhaustive.

3.4.4 Meaningfulness of Crowd Connections

We randomly sampled 727 crowd generated connections to inspect in detail. Apart from categorizing the crowd connections into three categories, our informal analysis found that 586 crowd connections represented meaningful facts (T1 or T2 connections) from the given context, even if they didn't match the gold standard connections. We were pleased to see that some crowd workers made reasonable speculations with the given information, and made use of their domain knowledge relevant to the context slice. For example, a crowd worker recognized a surname to be Arabic and made connections based on this domain knowledge. Some workers also used connection descriptions to suggest causation and pose hypotheses. For example, a worker connected "21-Apr-03" and "$35,000", describing their relationship as "After receiving this money more suspicious activity started on this day". Although this description did not strictly align with our task instructions, it illustrates the crowd's capability and willingness to provide more advanced and subjective insights.

We also found 141 connections that weren't meaningful (T3 connections). These issues may be dealt with by providing more specific instructions or style guides for workers, and by enhancing the interface to detect frequent mistakes. Several classes of such mistakes are already apparent from our experiment. For example, a worker connected two person names and described their relationship as "[these are] both names" or "[these] appeared in the same report". Another worker labeled a connection as "date they called this city". The system could ask crowd workers to avoid using pronouns or repeating given entity names in their relationship descriptions, or automatically detect if the name of entity categories (e.g. "name", "location") appear in relationship descriptions and alert workers about possible mistakes.

3.4.5 Limitations and Future Work

We analyze the quality of crowd connections by comparing them to gold standard connections provided by the creators of the Crescent dataset. We caution that the gold standard connections alone are not sufficient to evaluate crowd worker's results. The solution given in the dataset is written with a global context and include high-level hypotheses that cannot be generated with only two local documents. These analyses revealed similarities between crowd and expert performance and other indications of value, but further research is needed to explore the impact of crowd connections on an expert analyst's sensemaking process. Additionally, we only used a subset of one dataset for our experiment; follow-up studies are needed to understand how larger datasets or other types of documents affect crowd performance.

3.4.6 Lessons Learned

Our findings include a typology of three types of crowd connections: contextual connections, common-sense connections, and collateral connections. We found that context slices composed of documents with overlapping entities lead to better analysis quality.

We compared the crowd connections to gold standard connections and found that crowd workers were able to connect 85% of entity pairs mentioned in the gold standard connections, where the missing ones either require information from more documents than given to each crowd worker, or are connected to the same identity but in another entity of different format. A majority vote with threshold can substantially improve the precision and recall values of crowd-generated connections.

We considered the value of the crowd-generated connection descriptions (edge labels). We found that the description labels written by different crowd workers on each connection

converge to a small number of keywords that are usually accurate and sufficient for analysts to understand the relationship between two entities without reading the original documents. The results show that the node (entity) degrees, often indicating entity importance, in crowd-generated graphs are similar to those in the graph built from the solution. Entity pairs that are connected in the solution are also more likely to be connected by most crowd workers.

This work explores the use of non-expert crowds in the sensemaking loop of expert analysts, by restructuring the dataset into context slices for each crowd worker, such that they can do independent, in-depth analysis. We reified this technique in a web application that visualizes the dataset and enables crowd-created connections. We experimented different slicing methods and their impact on the quality of crowd analysis. We also contribute ways of aggregating crowd-generated connections that support strategic information retrieval and schematizing by expert analysts in large-scale textual data analysis.

Chapter 4

CrowdIA Pipeline: Bottom-up Building Path

Research Question 1 explored how to extract and discover the key relationships between entities with crowds. However, the input data was assumed to be perfectly relevant documents and the named entities are pre-categorized, both require non-trivial work from previous sensemaking steps. To make this work more practical and applicable to a broader domain of problems, we need to connect crowd work in different sensemaking tasks with minimal expert intervention.

This brings us to the second challenge: enabling large-scale collaboration on highly integrated cognitive activities. The sensemaking loop [154] contains multiple inter-connected sub-loops. I address this challenge by modularizing the process into a sensemaking pipeline and expanding the concept of context slices to modularize the data in each step.

In this chapter, we describe the process and challenges involved in designing the pipeline, particularly focusing on issues central to provenance and hand-off within and between sensemaking components. Then we present the bottom-up building path of the pipeline, the evaluation, and discussion.

4.1 Design Process: Preliminary Studies

The above related work suggests two hard problems in crowdsourcing the entire sensemaking process. One problem (RQ1.1) involves identifying information needs (*Step Input*) and intermediate analysis results (*Step Output*) of each step in the analysis, such that each step has a clear goal and progresses along the sensemaking process while preserving provenance. A second problem (RQ1.2) involves distributing the work required to analyze each *Step Input* among crowd workers and meaningfully aggregate local results as the *Step Output*. Below, we describe our approach to addressing these problems, leading to the final pipeline design implemented in CrowdIA.

4.1.1 RQ 2.1: Identifying Step Inputs and Outputs with Individual Participants

The sensemaking loop is a "broad brush description" [153] of an expert's cognitive process of information transformation. The boundary of each step is not clear-cut and the expert might skip steps. In order to support large-scale distributed sensemaking, a robust pipeline has to explicitly represent each intermediate analysis result. Our first preliminary study sought to uncover the information needs (*Step Input*) and intermediate results (*Step Output*) of each step when individual analysts rigidly follow the sensemaking loop (RQ1.1).

We designed a series of prototype interfaces to sequentially guide individuals' exploratory text analysis and specified the intermediate analysis results at five different steps. These five steps were based directly on the forward arrows in Pirolli's sensemaking loop (labeled 2, 5, 8, 11, and 14 in Fig. 2.1): 1) *Search and Filter* relevant documents; 2) *Read and Extract* key information pieces; 3) *Schematize* information pieces into meaningful node-link graphs; 4)

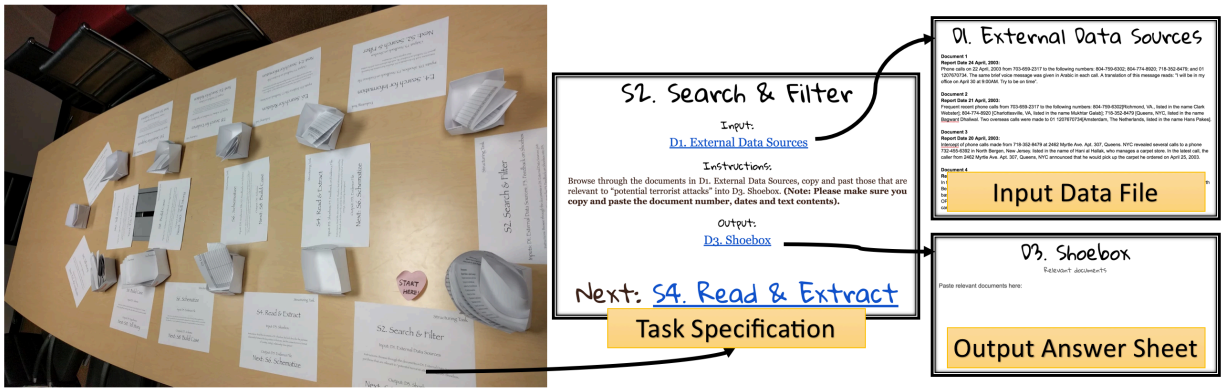

Figure 4.1: Prototype Interface: Example task specification interface, input data file, and output answer sheet.

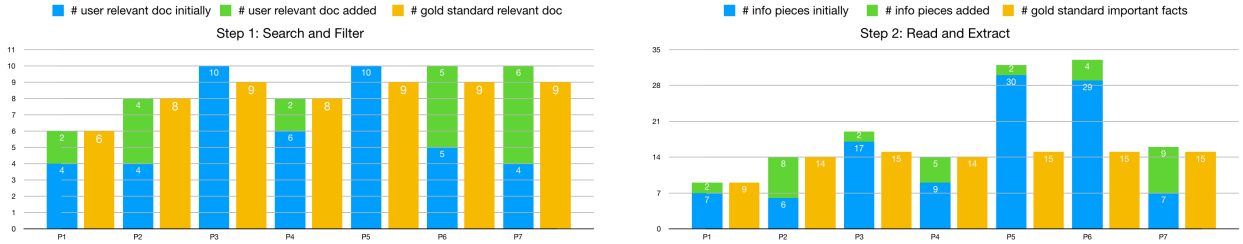

Figure 4.2: Intermediate analysis results by individual participants in Steps 1 and 2. Blue bars are their initial results, green bars are their second path edits, and orange bars are correct answers contained in each participant's results.

Build a Case with graphs to form hypotheses; and 5) *Tell a Story* with winning hypotheses in a narrative conclusion. The prototype served as a common artifact that participants could jointly author [62]. Each step was comprised of two information items (input data file and output answer sheet) and one task specification (Fig. 4.1). The input data file listed the contents of the input data. The output file was a blank sheet to fill in. The task specification gave the task name, instructions, the name of available input, the name of expected output, and the name of the next step. We designed the interfaces as lightweight prototypes [78] in Google Docs and Google Slides. Each participant completed Step 1 through 5, and then had the option to go back and refine their intermediate analysis in a second path edit. Participants attempted to solve mysteries using easy, moderate, and difficult data sets.

Provenance. Several participants found some of the sensemaking steps unnecessary when solving the easy mystery, while others appreciated the step-by-step pipeline to organize their thoughts and consolidate their analyses. All participants solving the difficult mystery needed to go back and refine their previous step output. Most participants added more documents and information pieces (Fig. 4.2), and modified their original node-link graphs (Fig. 4.3, left) after finishing the first path. They found the modularized steps helpful to keep track of their analysis process, and to trace back and improve certain intermediate results when refining their final conclusion.

Lesson : Modularized steps with clearly defined intermediate analysis results help ensure analysis credibility and support backtracking when refining the previous analysis.

4.1.2 RQ 2.2: Distributing Input and Aggregating Output with Crowd Workers

Given the *Step Input* and *Step Output* of each step, our second preliminary study explored how to distribute the *Step Input* among crowd workers and aggregate local analysis results into *Step Output* (RQ1.2). We deployed the same prototype interfaces on Amazon Mechanical Turk and added separate input and output interfaces for each crowd worker. After each step, we manually copied and pasted the crowd results to fill in the input interface for the next step.

Handoff Within Steps. We compared parallel and iterative human computation processes [126] for crowd collaboration within the steps. We found that a parallel approach allowed crowds to search and filter relevant documents (Step 1), but the same approach led to duplicated evidence extracted from the documents (Step 2). Workers rarely revised

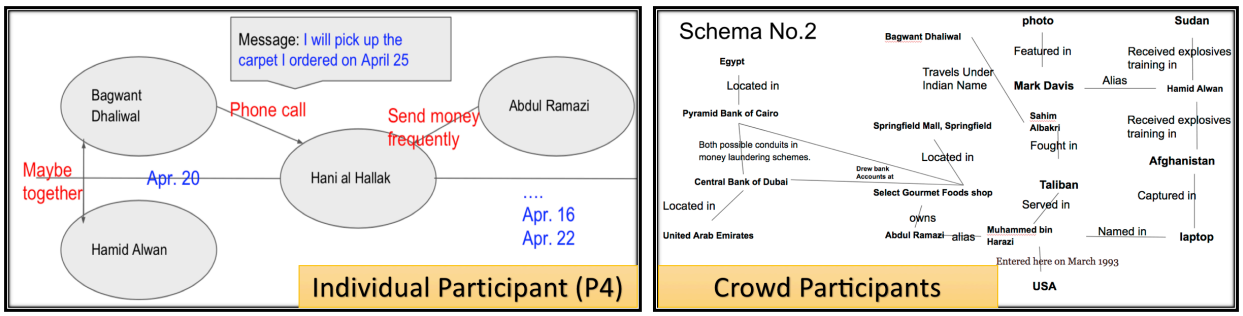

Figure 4.3: Example schemas created by individual participants (left) and crowd workers (right) in the preliminary studies.

previous schema or hypotheses created by others and tended to create their own new ones (Steps 3, 4). However, they refined and improved previous narrative conclusions written by other workers (Step 5).

Lesson : A parallel process works better for decision tasks, and an iterative process works better for information extraction and synthesis tasks. When new structures are introduced to reorganize the current information (Step 3), previous analysis results become difficult to understand and build on.

Handoff Between Steps. We observed that crowds were most challenged to understand schemas created by previous workers. The hypotheses these workers developed did not include key findings from the schema (Fig. 4.3, right) or had wrong or contradictory information (Step 4).

Lesson : The node-link graph is difficult for later crowd workers to understand since it does not provide an obvious starting point and represents many implicit, personal thought processes. We suggest a more effective approach to crowdsourced schematization is to use a less abstract and more well-defined structure, such as workers assigning appropriate pre-

defined tags to information pieces, which can then be organized accordingly.

4.2 The CrowdIA System

Guided by the lessons learned from the preliminary studies, we refined our pipeline and implemented a web-based application, CrowdIA, to automate its facilitation. Fig. 4.4 shows an example interface from the system.

4.2.1 Implementation

CrowdIA is implemented with the Django web framework and deployed on the Heroku cloud platform.

The back end is implemented in Python with a PostgreSQL database and uses the boto3 API to communicate with MTurk. It is responsible for 1) partitioning current step input into context slices; 2) sending context slices and corresponding contents to the front end when a worker accepts a task; 3) receiving and saving local task results (encoded as JSON strings) into the database when a worker submits a task result; 4) keeping track of local task status by detecting submitted, returned, or abandoned tasks; 5) aggregating local task results into *Step Output*; 6) keeping track of step completion status; 7) transforming current completed *Step Output* into the next *Step Input*; and 8) automatically releasing corresponding tasks to MTurk.

The front end is implemented with the Bootstrap UI framework in HTML, CSS, and JavaScript / JQuery. It is responsible for 1) rendering the UI design; 2) supporting user interaction (e.g., when extracting information pieces, a crowd worker is required to put the "who, what, where, when" elements of an information piece into four separate blanks); 3) validating re-

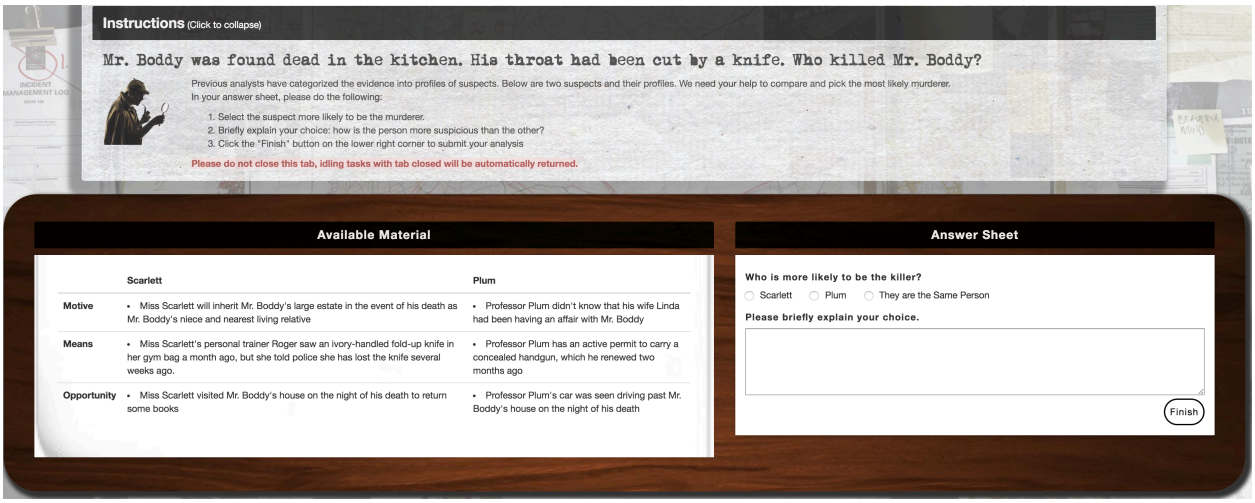

Figure 4.4: Example crowd worker interface for Step 4. On the top are task instructions including the global context (first line), task overview (first paragraph), and action items (bullet points). On the bottom left is one context slice as local task input (available material). On the bottom right is the local task output where crowds fill in and submit their analysis.

sults to ensure work quality; and 4) sending requests to the server to fetch task content and submit analysis results (via AJAX and JSON strings).

In the following sections, we first explain the overall pipeline structure, and then focus on describing the different input, output, context slices and aggregation mechanism of each step.

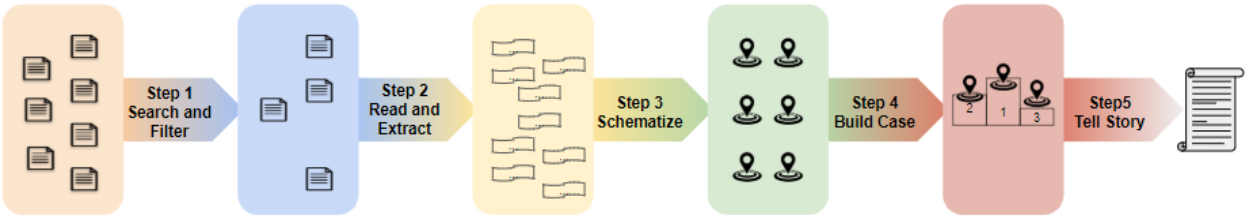

Figure 4.5: Modularized sensemaking pipeline. Step 1 searches external data sources for relevant documents. Step 2 extracts important information pieces from the relevant documents. Step 3 organizes information pieces into profile schemas. Step 4 compares and merges schemas to develop hypotheses. Step 5 synthesizes the best hypotheses as the final presentation.

4.2.2 Pipeline Structure and Step Definition

The CrowdIA pipeline is composed of five *Step*s, corresponding to the five data transformation processes in the sensemaking loop [153]. Each *Step* is a dedicated module defined by *Step Input* and *Step Output* (Fig. 2.1 right). Each *Step Output* equals the *Step Input* of the next *Step*.

Each *Step Input* is restructured and partitioned into multiple *Context Slices*, each of which is a meaningful subset of *Step Input* and contains semantically cohesive data. The *Context Slice Results* are aggregated into *Step Output* without further processing.

Each *Context Slice* is rendered in one or more *Local Tasks*, each of which will be assigned to one crowd worker. The results of *Local Tasks* submitted by crowd workers contributes to *Context Slice Result* via an *Aggregation Mechanism*. In this chapter, we implemented two commonly used aggregation mechanisms from the crowdsourcing community: majority vote and create-review [200]. The majority vote mechanism applies to rating, tagging, or voting tasks that are distributed in parallel among workers (Steps 1, 3, 4). A *Context Slice* uses the answer chosen by most of the workers (above a threshold) as the *Context Slice Result*. The create-review mechanism applies to free-text input (Steps 2, 5). The first crowd worker creates a free-text answer (one information piece or one narrative presentation); then, a second worker reviews and refines this result. The process continues until no new revisions are made. A *Context Slice* uses the final unrevised answer as the *Context Slice Result*. To ensure the quality of work, we ask crowds to provide brief explanations of their choices for the majority vote tasks; for create-review tasks, we provide self-assessment rubrics [53] below the free-text input boxes.

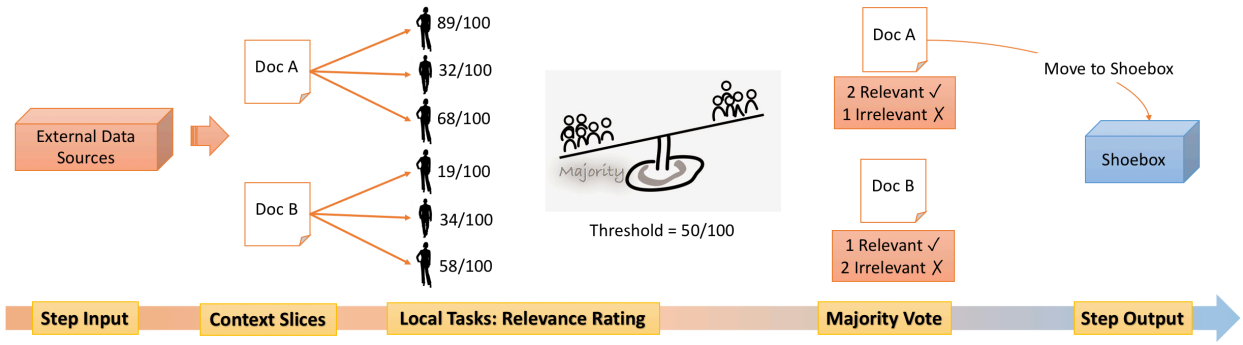

Figure 4.6: Step 1: Search and Filter. Crowds independently rate document relevance from 0 (completely irrelevant) to 100 (completely relevant). Using a predefined threshold, each relevance rating is converted to a binary vote. Documents with the majority vote will be passed to Step 2.

4.2.3 Step 1: Search and Filter

Step Input: All the raw text documents available for analysis.

Step Output: The subset of documents that are relevant to the global context.

Context Slices and Local Tasks: Each *Context Slice* contains n documents with shared entities, which could potentially "connect the dots" among documents and help workers better determine document relevance. Each *Context Slice* has no more than 600 words (2 minutes of reading for the average adult) and is rendered in $c \geq 3$ *Local Tasks*. Each crowd worker gives a relevance rating and provides a brief explanation.

Aggregation Mechanism: Majority Vote. A document is considered relevant if a majority of workers deemed it so. All relevant documents are put into the *Output* (Fig. 4.6).

4.2.4 Step 2: Read and Extract

Step Input: Relevant documents found in Step 1.

Step Output: A set of information pieces comprised of the key entities in the relevant docu-

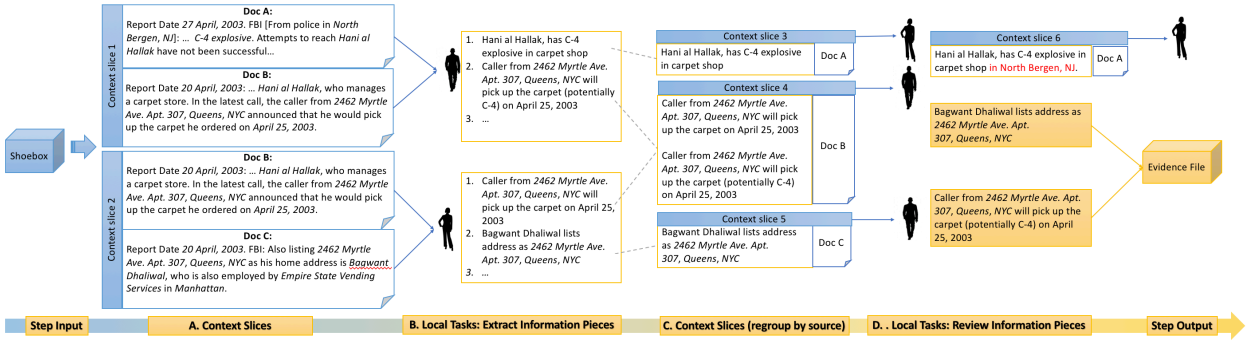

Figure 4.7: Step 2: Read and Extract. CrowdIA groups documents with overlapping entities into context slices of size $n = 2$ (A). The first batch of crowd workers extracts information pieces from context slices (B). The information pieces are then regrouped by their source documents into new context slices (C). The following batches of crowds review information pieces (D). The process continues until no new revisions are made.

ments.

Context Slices and Local Tasks: Each *Context Slice* contains n documents and (if not the first worker) information pieces extracted from the documents. For each *Context Slice*, $c \geq 2$ *Local Tasks* are rendered sequentially and the crowd workers extract or review the information pieces.

Aggregation Mechanism: Create-Review. Each *Context Slice* ends up with a list of final information pieces. Notably, when context slices have overlapping documents, crowds can extract information pieces that synthesize information from multiple documents. The disadvantage is that the same information pieces could be extracted multiple times by different workers. The reviewing process re-organizes information pieces by source documents into new context slices to help remove duplicates (Fig. 4.7).

4.2.5 Step 3: Schematize

Step Input: Information pieces extracted in Step 2.

Step Output: Tags on information pieces that identify targets and form a categorical schema.

Context Slices and Local Tasks: Each *Context Slice* contains n information pieces and is rendered in $c \geq 3$ *Local Tasks*, where each crowd worker fills in one free-text target and selects one or more predefined tags independently. The free-text target identifies candidates for the unknown element of the global task (e.g., potential target locations of a terrorist attack, or the suspect in a murder case). The predefined tags link each information piece to the known elements of the global task.

Inducing structure from new data with distributed crowds is a challenging problem [6, 28, 106]. In CrowdIA, predefined tags depend on the types of data and known facts. For example, we used "means," "motive," and "opportunity" tags for the moderate dataset because these three categories are common practices in determining the suspect in a criminal case. In the difficult dataset, we used all known elements as predefined tags because the target location is assumed to have the highest number of abnormal activities.

Aggregation Mechanism: Majority Vote. Tags that received a majority vote are retained for the information pieces in each *Context Slice*.Each free-text target has at least one information piece which also has at least one predefined tag. This creates a profile for each target candidate, which marshals the relevant information pieces into a tabular structure according to the predefined tags (Fig. 4.8). For example, each murder suspect's profile organizes all of their means, motive, and opportunity evidence.

4.2.6 Step 4: Build Case

Step Input: Target profiles containing information pieces tagged from Step 3.

Step Output: Preliminary hypotheses developed by comparing the target profiles.

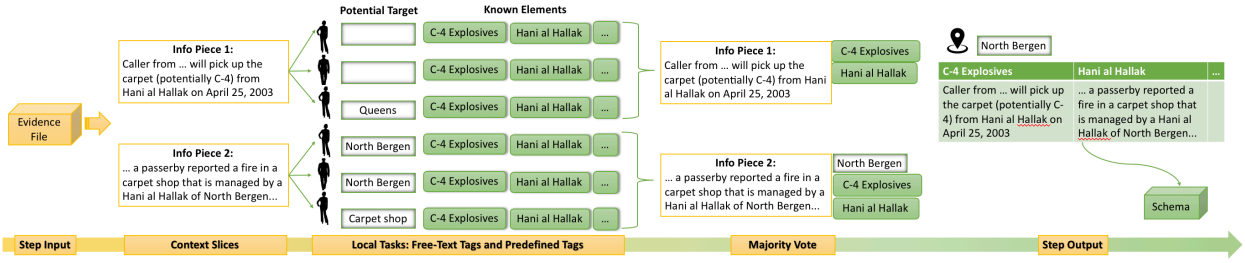

Figure 4.8: Step 3: Schematize. Crowds identify potential target locations and tag the information pieces with known elements. Information pieces are tagged with tags that earned the crowd's majority vote and organized into profiles of the candidate targets.

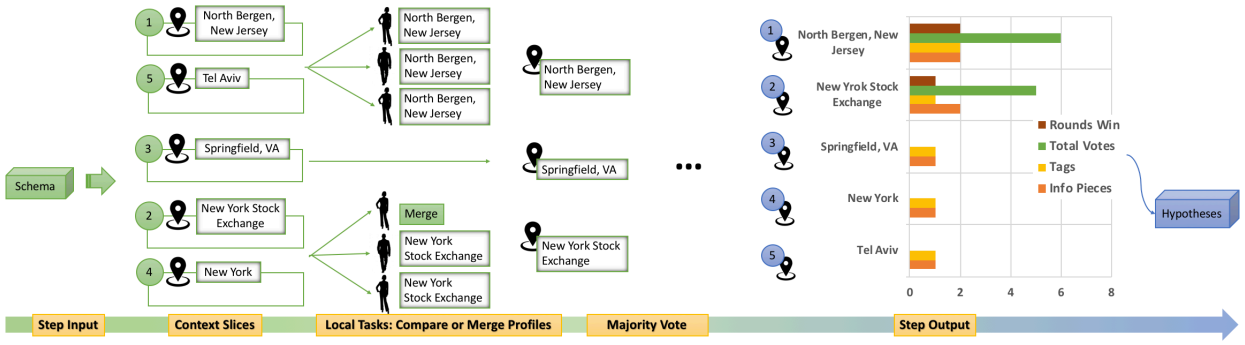

Figure 4.9: Step 4: Build Case. Crowds compare candidate profiles and merge aliases. As in a single-elimination competition, workers in Step 4 rank candidates by their perceived likelihood of being the target location.

Context Slices and Local Tasks: Each *Context Slice* contains n profiles and is rendered in $c \geq 3$ *Local Tasks*. Each crowd worker selects the most likely candidate or (in the case of aliases) declares them identical and provides a brief explanation. As a proof-of-concept, we adopt the single elimination tournament among candidates [67], each competition being a *Context Slice*. Profiles are initially ranked by the number of tags and information pieces (Fig. 4.9).

Aggregation Mechanism: Majority Vote. For each *Context Slice*, the profile with majority vote enters the next round of comparisons until only one profile is left.

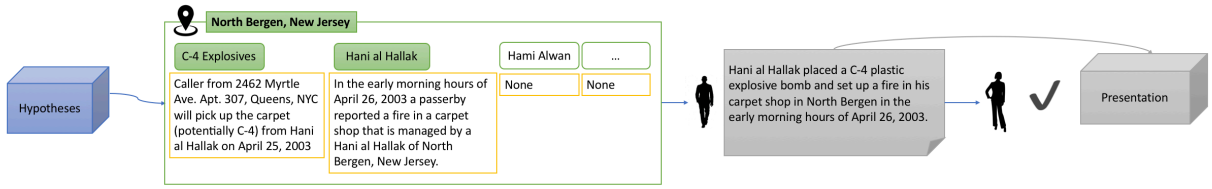

Figure 4.10: Step 5: Tell Story. Crowds put together the information in the winning profile and write a complete narrative. The presentation is ready when no new revisions are made.

4.2.7 Step 5: Tell Story

Step Input: The best preliminary hypotheses from Step 4.

Step Output: A narrative conclusion backed up by supporting evidence.

Context Slices and Local Tasks: Each *Context Slice* contains n profiles and is rendered in $c \geq 2$ *Local Task*s sequentially. Crowds write and review a narrative that integrates the profile information into a complete story (Fig. 4.10). In our experiments with one missing element (person or location), only the winning profile is considered the *Step Input*, so there is not really a need for a *Context Slice* when $n = 1$.

Local Task Aggregation Mechanism: Create-Review. Each *Context Slice* ends up with a reviewed presentation, which is also the output of the entire pipeline.

4.2.8 Refining Path: Top-Down

When the final presentation does not meet the expectation of clients, we need to refine the previous analysis. Furthermore, complex sensemaking like intelligence analysis is never fully complete, but rather becomes more valid with the available evidence [39]. As the analysis proceeds, new insights and lines of inquiry may arise. These issues motivate the needs to refine the previous intermediate analysis with the new knowledge learned and any feedback provided by clients.

There are two questions to ask before triggering the refining path: 1) What are the problems with the current analysis, and 2) From which steps did the problems originate? The first question decides if a refining path is needed, and the second question decides where and how the intermediate analysis should be refined. A reasonable start could be to ask an expert to review the current pipeline result, locate problematic intermediate results, and provide feedback for the corresponding step. It is also worth exploring how crowds can be leveraged in answering the two questions.

Once the step is identified and feedback is provided, new context slices are needed for crowd workers to address the feedback. We designed a feedback format with three elements: *context* (where is the problem), *critique* (what is the problem), and *task specification* (what is needed to fix the problem). The new *Context Slices* contain the formatted feedback and a subset of *Step Input* that produced the feedback *context*. Each *Context Slice* is then rendered in local tasks similar to the bottom-up path. Each crowd worker first evaluates whether the feedback can be addressed with the given material, then submits an explanation of why not or new results to address the feedback. The new results are aggregated into a refined *Step Output*. If the new *Step Output* is different from the previous one, the subsequent steps will be re-executed with the new analysis. Otherwise, the expert will need to find another previous step to fix.

4.3 Evaluation: Solving Mysteries with Crowds

To evaluate the feasibility of the pipeline, we deployed CrowdIA on Amazon Mechanical Turk (MTurk) and asked crowd workers to solve three mysteries of increasing difficulty levels. To evaluate the quality of crowd analysis with CrowdIA, we compare the conclusions by crowds to the correct answers of the mysteries. In this chapter, we focus on evaluating the bottom-up

forward pipeline, leaving experiments with the top-down refining path for future work.

4.3.1 Method

We recruited participants from MTurk US-only pool and paid at least minimum wage in our location ($7.25 per hour). Based on the time the individuals and crowds spent in the preliminary studies, as well as the effects of queuing and/or idling tasks, we determined the time needed to complete local tasks in each step as: Step 1: 1.7 min, Step 2 extract: 2.8 min, Step 2 review: 2.5 min, Step 3: 1.6 min, Step 4: 3.3 min, Step 5 create and review: 2.2 min. Consequently, to ensure at least minimum wage, we provided the following payments for the local tasks: Step 1: $0.20, Step 2: $0.34 and $0.30, Step 3: $0.19, Step 4: $0.40, Step 5: $0.24 and $0.24. CrowdIA posts each step on MTurk as a Human Intelligence Task (HIT) that dynamically renders the context slices and assigns a worker to each slice. We assign each worker to only one context slice of one step to demonstrate the capability of distributed novice crowds to solve mysteries and mitigate learning effects or collusion. Crowd workers who quit an accepted HIT without submitting it were not allowed to resume the unfinished work or take a new HIT.

We evaluate our pipeline with easy, moderate and difficult datasets (for details on the datasets, see Appendix A.1). We consider datasets with more documents, more elements (who, what, where, when) and more complicated relationships among elements to be more difficult.

4.3.2 Results of Easy Dataset

The easy dataset contains three documents about three girls who might have ruined Mr. Potter's flowerbed [51]. The crowd workers successfully found the culprit and presented

their conclusion with supporting evidence.

We recruited five crowd workers, each working on one step in the pipeline. The first worker took 34 seconds to find the two relevant documents, the second took 12.5 minutes and extracted 10 important information pieces, the third took 22.7 minutes and organized the information pieces into four groups, the fourth took 10.1 minutes and generated three hypotheses for each suspect, and the fifth took 5.7 minutes to pick the most likely culprit, offering the following conclusion:

> I think it was Serina who had the muddy shoes after playing hopscotch. Her shoes were muddy so that could indicate that she went into the just watered flower bed. Maybe she only ran through it to get to her friends so they could play but she might have stomped on the flowers on her way to the play area. She was in a hurry and not paying attention to what she was doing.

Detailed crowd analysis results for the easy dataset are presented in Appendix A.2.

4.3.3 Results of Moderate Dataset

The moderate dataset, inspired by the popular Clue board game, has nine documents; three suspects with different means, motives, and opportunities to kill Mr. Boddy; and four witnesses. The crowds successfully identified the murderer and backed up the conclusion with supporting evidence.

We recruited a total of 76 crowd workers to analyze the dataset. In Step 1, 27 workers found seven relevant documents and excluded the two documents about wrong means and wrong opportunity of the two wrong suspects (time spent in minutes: mean=5.8, median=2.9, std=8.78). In Step 2, 14 workers extracted eight information pieces, of which seven were

important (time spent on creation tasks in minutes: mean=10.7, median=4.6, std=18; time spent on reviewing tasks in minutes: mean=7.6, median=3.81, std=9.61). In Step 3, 24 workers tagged the seven important pieces with person names and means / motive / opportunity evidence type, whereas the useless information piece received a None tag (time spent in minutes: mean=7.7, median=3.5, std=12.79). One of the witnesses also got tagged as a potential suspect, resulting in four profiles. In Step 4, nine workers weighed the suspect profiles in the single elimination tournament and Scarlett was deemed the most likely murderer (time spent in minutes: mean=13.0, median=7.2, std=14.92). Finally, in Step 5, 2 workers narrated the conclusion (creating worker took 23.7 minutes, reviewing worker spent 2.05 minutes):

> Miss Scarlet killed him [Mr. Boddy]. She was seen at Mr Boddy's house on the night of his death. She also had the murder weapon seen by her trainer in her bag. Also, she had the motive since she would inherit his estate.

Detailed crowd analysis results for the moderate dataset are presented in Appendix A.3.

4.3.4 Results of Difficult Dataset

The difficult dataset is part of the *Sign of Crescent* dataset [89] used as training material for professional intelligence analysts. We streamlined the Crescent dataset to only cover one terrorist plot, added extra documents as noise, and specified the goal as identifying the target location of the attack. There are 13 documents, four terrorists, and 12 locations mentioned in the documents.

We recruited a total of 135 crowd workers: 18 workers in Step 1 (time spent in minutes: mean=10.9, median=4.3, std=15.6), 22 workers in Step 2 (time spent in minutes:

mean=17.3, median=10.7, std=17.2), 78 workers in Step 3 (time spent in minutes: mean=12.1, median=3.1, std=16.0), 15 workers in Step 4 (time spent in minutes: mean=9.9, median=7.3, std=9.0), and two workers in Step 5 (creating worker took 47.4 minutes, reviewing worker spent 1.5 minutes).

Echoing the results from our preliminary study, the crowd workers were one step away from the actual target location New York Stock Exchange. However, they found the weapon storage location North Bergen, New Jersey and ranked New York Stock Exchange as the second possible target. Below, we examine the crowds' performance in detail.

Step 1: Crowd successfully retrieved indirectly relevant documents. In Step 1, seven documents out of 13 directly mentioned one or more key elements and are automatically considered relevant. The remaining six documents (three indirectly relevant and three irrelevant) are each rated by three crowd workers on a 0–100 scale. The crowds found four relevant documents (with one extra irrelevant document) from the six documents with a threshold of 50, resulting in 11 relevant documents. A follow-up analysis found that thresholds ranging from 30 to 60 would lead to the same result, with lower thresholds increasing false positives and higher ones increasing false negatives. Thresholds 0–10 would include all three irrelevant documents, 15–25 would include two irrelevant documents, 65–70 would include one irrelevant document and miss one relevant document, 75 would include one irrelevant document and miss two relevant documents, and above 80 would not include irrelevant documents but miss three relevant documents.

Step 2: Crowd extracted most key useful information pieces. In Step 2, a total of 26 information pieces were extracted from the documents, of which 18 were useful ones. The information pieces cover key evidence about terrorists' real names and aliases, phone calls, and the bomb and the storage location. We found that the crowds were able to synthesize information across two documents, viz.: "Hani al Hallak's carpet shop in North Bergen caught

fire" and "Police found C-4 explosives in the carpet shop reported on fire in North Bergen" were extracted as one information piece: "Hani al Hallak's carpet shop has C-4 explosives." The crowd's review process solved issues like misspellings, incomplete name references, missing elements (who, what, where, when, etc.), and duplicates.

On the other hand, not all important information pieces were extracted. One important information piece showing that one of the terrorists works in the actual target location did not get extracted. Some information pieces about the relationships and roles of terrorists also did not get extracted. The missed activities were cover-ups of terrorists and not obvious to an early-stage investigation. These issues could be improved by re-executing Step 2 with additional feedback.

Step 3: Majority vote elicits accurate tagging and potential target identification. After Step 3, 18 of the 26 information pieces were tagged, excluding the four information pieces from the irrelevant document. Following the majority vote aggregation, all information pieces were accurately tagged with the key evidence.

We closely examined the tags and found that individual crowd workers tended to give information pieces more tags than strictly needed. Some workers just selected every tag and others selected nothing when they could not identify any locations. The majority vote mechanism helped eliminate the influence of such low quality work and only kept the accurate tags.

Five location tags were created. One notable development was that two different workers both identified a location "Tel Aviv" in the information piece: "I will be in my office on April 30 at 9:00AM. Try to be on time." One of the workers even gave very specific information: "the location is Israel at Mike's Place, a restaurant in Tel Aviv." We later learned that there was a real "Palestinian suicide attack perpetrated by British Muslims which killed three civilians

and wounded 50 at Mike's Place in Tel Aviv on April 30, 2003" [1], the same timeframe as the dataset. Although crowds were instructed not to add extraneous information, these two workers aligned the information in the given context slice with their external knowledge and mental model.

Step 4: Crowd logically reasoned and weighed hypotheses. The ranked location tag results are shown in Fig. 4.9. The final winning location was North Bergen, New Jersey, the last place the bomb was stored before transferred to the target location. The runner-up, losing by only one vote, was the correct answer, New York Stock Exchange. Even though the crowd narrowly missed the actual target, the winner is the second-most crucial location to investigate. The correct answer, New York Stock Exchange, was merged with another location, New York, and won one of the competitions with insightful explanations by workers:

> The New York Stock Exchange is a specific, high value target for terrorists because a bomb attack there would likely cause many casualties and have a negative effect on the US economy. Springfield, VA is a very broad target and besides the fact that one of the terrorists lives there there isn't much evidence than an attack will take place there.
>
> — *Worker 1, New York Stock Exchange vs. Springfield, VA (Round 2)*

> There are multiple pieces of evidence showing suspicious activity centered on the NYSE. There's just one pieces of evidence pointing to Springfield, and it's just that a suspect lives there, there's no real evidence he's doing anything there.
>
> — *Worker 2, New York Stock Exchange vs. Springfield, VA (Round 2)*

Unfortunately, New York Stock Exchange lost in the final competition with North Bergen, New Jersey, where the terrorists store the bomb in a carpet store before transferring it to New York. However, the explanations were not as insightful or convincing, e.g.:

> They found an actual C4 in New Jersey, which makes me believe that was more likely meant to be the target.
>
> — *Worker 3, New York Stock Exchange vs. North Bergen, New Jersey (Round 2)*

Step 5: Crowd wrote a clear narrative presentation. Using the profile of North Bergen, New Jersey, workers from the last step created a narrative that connected the evidence to current findings and justified the likelihood of this place being a potential target. The final presentation created by the crowds was: "Hani al Hallak placed a C-4 plastic explosive bomb and set up a fire in his carpet shop in North Bergen in the early morning hours of April 26, 2003."

4.4 Discussion

4.4.1 RQ 2.1: How can we modularize the sensemaking process?

Analysis provenance enables step-wise debugging of the sensemaking process. Modularizing the sensemaking steps with explicit definitions of information needs (inputs) and intermediate analysis results (outputs) enables step-wise debugging and refinement, breaking down a big black box into smaller, more inspectable modules. For example, when the crowds analyzed the difficult dataset, they failed to extract some important information pieces (false negatives) in Step 2. This resulted in incomplete profiles of the potential target locations, which we believe led to the narrow miss of the correct target. By examining the intermediate analysis results, either experts or crowds could have potentially debugged the situation and refine the analysis in a top-down refining path (as in Figure 4.5). In future work, experts could manage the execution of the pipeline, similar to CrowdWeaver [**?**], and provide structured and situated feedback [116] for crowd workers to refine previous analyses.

Alternatively, crowds themselves could critique and refine intermediate analysis results in a feedback pass. In prior work, crowds have been used to provide feedback [132, 193] on visual designs, accurately evaluate each other's credibility [185], and react to personalized expert feedback while brainstorming [24].

Modularization enhanced scalability, resusability of analysis, and efficient division of labor. CrowdIA enabled as many as 134 transient novice crowd workers to collaborate to solve a difficult mystery, producing high-quality, insightful analysis output. The same number of people working together in a collocated way would be a big challenge to coordination and communication. Getting the same number of trained analysts working at the same time would be even more difficult. CrowdIA's automated facilitation mitigates logistics burdens, enabling workers to invest their time and efforts in the core analysis tasks. Later crowd workers continued the analysis from where previous workers left off, without requiring the previous workers to explain their intermediate results or thought processes. Furthermore, CrowdIA did not require crowds to have significant sensemaking expertise. Workers were all novices and transient, without prior exposure to the dataset, and giving only a small time commitment (typically a few minutes) each. These features can open up the sensemaking process to dynamically recruit from a much bigger pool of contributors.

Alternative strategies to schematize information. Steps 3 and 4 could be modified to support many different types of schemas. Different structures could be helpful for different types of analyses [64]. A node-link graph structure is very general to capture many types of relationships in the data, but can be difficult to hand off during collaboration [198]. CrowdIA implemented a more specific tabular structure to represent suspect profiles, which resulted in accurate hypotheses.

In CrowdIA, we implemented both location-centered (difficult mystery) and person-centered (easy and moderate mysteries) profile schema strategies. Alternative methodologies, such as analysis of competing hypotheses (ACH) [87], could also be applied. We found that an effective strategy is to tag information with appropriate categories with which the information pieces can be organized from different perspectives. This strategy is simple for novice crowd workers and highly scalable. Future work can also explore a data-centric approach by inducing tags directly from the documents [6, 106].

Optimizing each step with best-suited techniques. The modular design can support future research on optimizing each step of the pipeline as well as the overall workflow. For information foraging stages (Step 1 and 2), advanced algorithmic techniques [164, 180] can be leveraged to improve efficiency. Crowds can focus on edge cases where machine learning models do not perform well [28], which in turn helps train the machine learning models. For information synthesizing tasks, crowds are better suited than algorithms and have shown success in other applications [81, 134]. More complicated approaches like online contest webs [134] can be applied to guide the crowds to build hypotheses. Experts can also take over whenever they deem it appropriate.

4.4.2 RQ 2.2: How do we distribute and aggregate the analysis in each step?

Decomposing a big problem into small manageable problems has been a major challenge for the crowdsourcing community. To make sense of large amounts of data, many solutions employ a single step for sensemaking and distributing the work by 1) showing each worker all the data, 2) showing each worker one piece of data, or 3) showing each worker an arbitrary subset of data. All such microtasks are linearly defined, similar to how we divide the

Step	Context Slicing Goal	Context Slices	Alternative Slicing Methods
1	Provide context to define relevance	Documents that share entities	Documents of the same topic, sources
2	Provide context to complete missing parts of facts	Information pieces from the same source documents	Information pieces created by the same people, date
3	Provide context to identify evidence type	Information pieces that share entities	Information pieces with similar relationship types, connotation
4	Provide context to combine and/or compare schema	Profiles of suspects organized by evidence types	Evidence types organized by date, location
5	Provide context to back up each hypotheses statement	Most likely suspect and complete profile	Strongest suspects for each evidence

Table 4.1: Customized context slicing of each step depends on the level of analysis and the goal of the step.

documents in the first step. Instead of naively passing uniform local tasks from Steps 1 to 5, we create *Context Slice*s that divide each *Step Input* into cohesive subsets, and aggregate the *Context Slice Results* into *Step Output* (which is also the next *Step Input*), before creating new context slices.

Context slices enable meaningful and scalable division of work. Although the steps in the sensemaking pipeline were already modularized for experts, our concept of *Context Slice*s partitions the each *Step Input* so that novice crowd workers can contribute meaningfully. *Context Slices* enable workers to generate meaningful results that synthesize information beyond what can be extracted from a single piece of information or an arbitrary subset of information. For example, in the first step of solving the difficult dataset, when given a context slice containing one directly relevant document and one unrated document with shared entities, the crowds were able to identify other indirectly relevant documents. Without context slices, workers in the preliminary study were not able to identify these indirectly relevant documents.

Context slice design depends on data and differs among steps. An open challenge is that context slicing methods must be carefully designed for each step. CrowdIA implemented customized context slicing methods for each step in the pipeline (Table 4.1). We specify the context unit (how the slices are defined) according to the level of analysis in each step: documents, information pieces, and profiles, and the context slicing goal (how the slices are determined) according to the step goal. Exploring the trade-off between the size of context slices and the quality of local task output, and exploring alternative context slicing methods, are both promising directions for future work.

4.4.3 RQ 2.3: How do crowds perform in solving mysteries with the modularized pipeline?

Handling false positives and false negatives. Crowds handled false positives within the pipeline. For example, in the difficult dataset, crowds included one irrelevant document (a false positive) in Step 1, which propagated the useless information to later steps. However, Step 3 guaranteed that only useful information pieces are tagged with evidence types and put into profiles of candidates. Thus, the useless information was filtered out in Step 3. The pipeline was able to recover from the false positive and save worker labor in later steps.

We did not encounter false negatives in Step 1 with a rating threshold of 50. In general, the trade-off between false positives and false negatives could potentially be controlled by the rating thresholds [121]. We encountered false negatives in Step 2, where the crowds failed to extract all the relevant information pieces. The bottom-up pipeline alone did not recover from the false negatives. Future work on the top-down refining path could help resolve false negatives.

Crowds used external knowledge in information foraging. We found that sometimes crowds connected their own knowledge to the dataset. In Step 3, the workers created a location tag "Tel Aviv" which is not mentioned anywhere in the documents. This connects the information from the documents under investigation to external knowledge from the crowds. Despite our assumption (Appendix A.4) that no external knowledge is needed to solve these mysteries, the wisdom of crowds potentially broadens the coverage of the investigation.

Crowd explanations provided diverse perspectives in information synthesis. In the synthesizing stages (Steps 3–5), we found that the crowds provided diverse perspectives. When comparing suspects in the moderate dataset, one crowd worker chose the wrong suspect (Professor Plum instead of Miss Scarlett), and provided the explanation: "To cut a man's throat you would need to be at least as strong as him, I don't think women in general have the same sort of physical power as men, therefore I don't think Miss Scarlet had the physical strength to overpower Mr. Boddy and cut his throat..." Although this hypothesis does not align with the correct answer, real-world investigation can benefit from such insights for further data collection and analysis. Further exploration on collecting, structuring and making use of crowd explanations would be valuable future work.

4.4.4 Generalizability

We chose to deploy our pipeline to solve intelligence analysis mysteries because they exemplify the challenge of exploratory analysis: building robust and logical hypotheses from known facts to achieve a final conclusion as close to the hidden truth as possible. However, we envision the pipeline as adaptable to broader applications with different sensemaking challenges, as well as opening up more in-depth research within each step.

Broader applications beyond intelligence analysis. The general class of "mysteries" CrowdIA may help solve is potentially broad, including investigations in law enforcement, journalism, and human rights advocacy. However, we also expect that our pipeline can be adapted for other sensemaking tasks and domains. While future work is needed to understand the trade-offs, we anticipate that our approach will translate most directly to sensemaking tasks that involve uncovering hidden patterns or relationships among many text-based documents, such as coding qualitative data [6] or synthesizing creative ideas [26]. Our approach may also be suitable for sensemaking tasks that incorporate personal preferences, such as trip planning [197], online shopping [106], or researching home improvement solutions [81], because our pipeline already assumes iterative cycles of client feedback and revision. Finally, our approach may support crowdsourced sensemaking to generate hypotheses of biological and environmental phenomena [129, 151]. For these latter problems, it may be necessary to modify CrowdIA's steps to better align with the scientific method rather than the sensemaking loop of intelligence analysts.

Flexible crowd compositions and collaboration settings. A major constraint of using crowds on MTurk is that workers are transient and novices. This serves well for our purpose as a proof-of-concept, but in real-world intelligence analysis, the crowd's analysis may serve as an assistance to experts. When solving the difficult dataset, crowds were able to prune the noisy information from the documents without much loss of important information. Expert analysts could focus on the pruned information for more advanced analysis. Along with being a more efficient division of labor, this approach also allows for professional oversight, preventing novice crowds from jumping to wrong conclusions that can result in harmful consequences.

When confidentiality is a concern, one possibility is to incorporate task assignment techniques

for sensitive documents [23], but these may limit workers' access to global context and degrade quality. Another possibility is to use a trusted internal group who can access the confidential documents when collocated, synchronous, devoted experts are not available. Many data management businesses already employ or have access to such internal worker pools [137].

4.4.5 Limitations and Future Work

In this chapter, we focused on first establishing a proof-of-concept pipeline that orchestrates crowdsourced sensemaking, and then investigating how well the pipeline can facilitate novice asynchronous distributed crowds in solving mysteries. However, we did not empirically compare our approach to other sensemaking techniques or systems. Future evaluation studies could compare CrowdIA's highly structured process to more free-form approaches (e.g., [72, 170]) to articulate the trade-offs of exploiting Pirolli and Card's sensemaking structure. Comparing to alternative data slicing techniques, such as 1) extreme slicing, in which each worker gets only one document and votes for the likely target based entirely on local information; or 2) no slicing, in which each worker sees all the documents and attempts to solve the mystery, could suggest pipeline modifications that enable greater flexibility in worker time commitments and microtask granularities. CrowdIA's modular approach also suggests opportunities to experiment with alternative task designs for specific steps within the proposed pipeline, using this chapter's configuration as the baseline, similar to the experimental framework organized by Parikh et al. [152] for computer vision research.

A challenge in conducting controlled studies of CrowdIA is cost. Running the pipeline to solve the moderate and difficult mysteries required approximately 100 crowd workers and cost $50 per execution. Although other crowd-based systems, especially those requiring workers

with specialized expertise, can be much costlier [81, 134, 175], CrowdIA executions will become more expensive as the mystery gets more complicated, to compensate the increasing numbers of workers. Alternatively, researchers could leverage public enthusiasm for solving mysteries [146] to recruit volunteer crowds. These self-selected participants may bring more dedication and expertise to the problem, but may be less motivated by the artificial data sets common to controlled experiments.

To manage the scope of the problem, we enforced some key assumptions (see Appendix A.4) on the initial data input and the final result output associated with the target mysteries. Further field research is needed to understand how to relax these assumptions when applying the pipeline to more complex, real-world mysteries.

Chapter 5

Challenges in the Bottom-up Pipeline

The building path of the pipeline modularizes the bottom-up sensemaking process to support crowds in developing a theory from the raw big data set. It defines the inputs and outputs, as well as the context slicing methods for each step. The evaluation in Chapter 4 shows impressive success in crowdsourced mystery solving with the pipeline. Will the crowd always perform so well? Understanding the consistency in crowd performance is important for further optimization of the building path as well as designing the top-down refining path to develop incremental solutions with new findings from the building path. In this chapter, I will describe my evaluation of the building path of the pipeline in terms of the challenges faced by the crowds in each step.

5.1 Methods

5.1.1 Participants

We deployed the pipeline on Amazon Mechanical Turk (MTurk) with workers of higher than 90% approval rate. This is a relatively lower requirement compared to most of the similar crowdsourcing research. For example, Crowd Synthesis [6] used 95% approval rate on MTurk with additional training, and flash teams used crowds of experts from Upwork [176]. Our goal is to involve crowd workers at a larger scale and make our results more generalizable

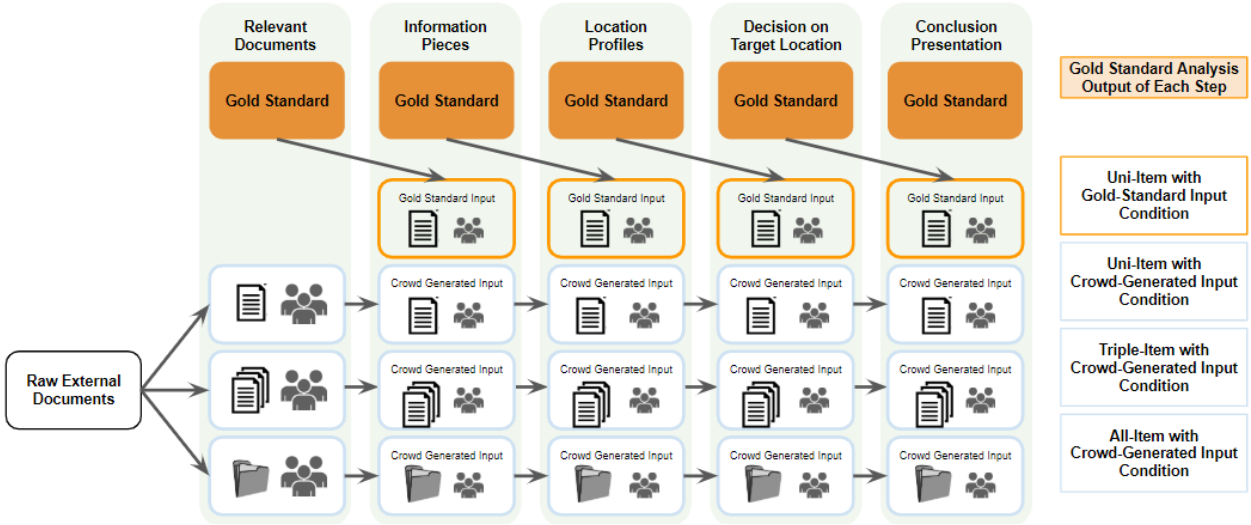

Figure 5.1: Experiment Design: Besides the five different steps in the pipeline, we manipulate the quality of step input (gold-standard or crowd-generated) and the size of context slices (1, 3, or all items in the step inputs). There are 4 conditions in total.

to different real-world problem-solving situations. We estimated the time needed for each microtask based on pilot studies and paid a fixed amount for each Human Intelligence Task (HIT) with the minimum wage of our location ($7.25 per hour).

5.1.2 Task and Procedure

Our experiment aims to investigate the crowd competency in different sensemaking tasks, and probe the source of errors by manipulating the quality of step input (gold-standard or crowd-generated) and the size of context slices (1, 3, or all items in the step inputs) (Fig. 5.1).

5.1.3 Context Slicing Methods and Choices.

In the uni-item condition, each item (documents, info pieces, profiles) in the step input is considered a context slice, so the number of context slices equals to the number of items in

the step inputs. In the all-item condition, the entire step input is considered as one context slice, so the number of context slices is always one.

In t theriple-item condition, the ways the items are distributed among context slices of size 3 is more complicated. Taking Step 1 as an example, there are $\binom{15}{3} = 455$ possible combinations to distribute the 15 raw input documents into context slices of size 3. As a proof-of-concept, we implemented a context slicing method that favors item similarity defined by entity overlap and does not allow overlapping items between context slices. This is to guarantee that the information is evenly distributed among workers and each input item is analyzed exactly once. If some documents appear in more context slices, they will be analyzed in more microtasks and by more workers. As found by Willett et al. [187], this can introduce biases favoring the information in more heavily analyzed documents.

5.1.4 Pipeline Execution Walkthrough

One execution of the pipeline results in one batch of mystery-solving analysis from steps 1 through 5. For each batch, the system starts with the 15 documents and executes the five steps sequentially.

Step 1: rate document relevance. The step input is always the 15 raw documents in all conditions. For each context slice, the system assigns 3 crowd workers to rate the relevance from 0 (completely irrelevant) to 100 (completely relevant) of each document in the slice. Ratings above 50 (neutral) are considered as positive. Each worker is also required to briefly explain their rating rationales in a text box. Documents with a majority vote for positive relevance are considered as relevant.

Step 2: extract important info pieces from relevant documents. For each context slice, the system first assigns 1 crowd worker to extract all the important info pieces from the

documents in the slice. The crowd was instructed to format info pieces as simple sentences structured as "who, what, where, when" as much as possible. After that, documents in each context slice and the extracted info pieces are assigned to a second worker for review. Reviewers are instructed to correct errors in the existing info pieces, delete any bad or useless ones, and add new ones to include any missing information.

Step 3: tagging info pieces. For each context slice, the system assigns 3 crowd workers to tag the possible target locations mentioned (if any) and the related known facts in each info piece. Each worker is also required to briefly explain their tagging rationales in a text box. For each info piece, workers will select one or more tags. Tags with a majority vote will be attached to the info piece. Info pieces that are tagged the same location names are all put together to form a *location profile*, and organized by their evidence types.

Step 4: evaluate location profiles. For each context slice, the system assigns 3 workers to evaluate the profiles in it. When context slice size is 1, each worker rates the likelihood of the location to be the attack target, from 0 (completely unlikely) to 100 (completely likely). Ratings above 50 (neutral) are considered as positive. Each worker is also required to briefly explain their rating rationales in a text box. The location with the highest average rating is considered as the most likely. Otherwise, the workers pick the most likely location in a given context slice and explain the rationale in a text box. Locations with the majority vote are considered as more likely than the others in each context slices. The process repeats until only one most likely location is left.

Step 5: write a narrative presentation to summarize the conclusion. Step 5 only works on the most likely answer thus is the same across conditions. For the most likely location, the system first assigns 1 crowd worker to write a narrative story that explains why the given location is most likely the target of the attack. After that, the story and the winner profile are reviewed by a second worker. In reality, Step 5 only needs to run once with two

workers (one writer and one reviewer), but for the purposes of this chapter, we ran it 4 more times ($4 \times 2 = 8$ workers) to gather more data and be comparable to the other conditions (Table 5.1).

5.1.5 Data Analysis

The data that informs this analysis includes the microtask responses submitted by the workers; the step outputs aggregated by the system; the system log of workers previewing, abandoning, and submitting the tasks; and the login/logout time for each worker.

We first compared the crowd analysis to the gold-standard analysis. Except step 1, all other steps with crowd-generated input require a qualitative comparison between the crowd output to the gold-standard output. Specifically, for the info pieces extracted in step 2, the crowd might extract the same information in different ways. We coded for two levels of correctness: 1) *matching* the gold-standard info piece, and 2) *not matching but relevant* and useful to solving the mystery. Since the crowd might partially extract the info pieces, we also count the number of matched and relevant elements in the crowd results. An element is any one of the "who, what, where, when" items in the info piece. In step 3, we compare the resulting location profiles to the gold-standard analysis. We also qualitatively examine the tagging results and explanations by the crowd. In step 4, we manually rank the crowd-generated locations with the same criteria as used when ranking the gold-standard locations, then compare with the crowd rankings. In step 5, we examine the number of retrieved key evidence and qualitatively evaluate the writing by the crowd.

In addition, we open-coded and analyzed the crowd explanations from steps 1, 3, and 4, identifying common behaviors and speculating on crowd analysis rationales. Two authors first sampled around 10% of the data and analyzed separately, then compared the coding

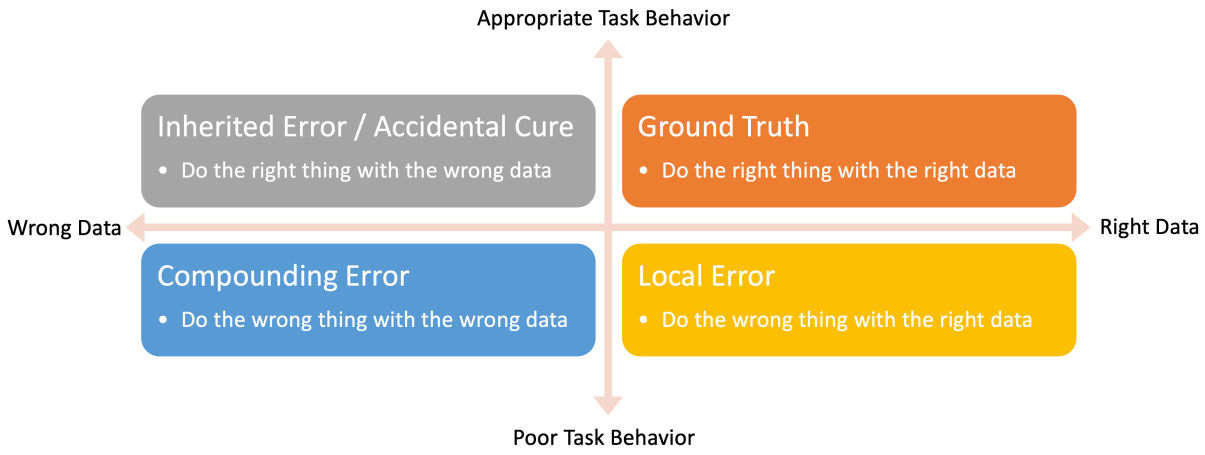

Figure 5.2: Task performance measured by two sources of errors: data quality and task behaviors

to agree on a set of codes with clear definitions. After that, author A focused on comparing the crowd results to the gold-standard analysis, while author B focused on coding the explanations provided in steps 1, 3, and 4. The two authors then reviewed and iterated on the analysis until reaching a consensus. From the task performance perspective, we categorize the source of error with respect to data quality and task behavior (Fig. 5.2). We classify the data correctness by comparing to gold-standard analysis. We define "the right thing" in task behavior by the following four possible levels of analyses:

- Accurate: true to the information source (directly copied from the document text)
- Focused: relevant to the investigation goal
- Interpretive: rephrase what the facts *mean* (not directly copying)
- Deductive: synthesize facts and develop hypotheses (including facts from multiple documents and hypotheses not directly mentioned in any document)

5.1.6 Limitations

Our evaluation studies have several limitations. We focus on one specific pipeline and software, one crowdsourcing platform with one recruiting requirement, and one example data set. These choices might limit our generalizability. However, the pipeline modularizes the mystery solving process into representative subtasks, and results on local tasks resonate with prior work. Future research should explore the pipeline application in different types of problems and scenarios.

While the pipeline is adapted from the sensemaking loop that is widely used in sensemaking research, including crowdsourced sensemaking, there are alternatives, such as data-frame theory [109], that are also prevalent. It is possible that the crowd shows different analysis performance with a different underlying theory and framework.

Additionally, our participants are recruited from Amazon Mechanical Turk with a low requirement of 90% approval rate. Workers with different approval rate or from different platforms, such as expert crowds from Upwork or volunteers from Reddit, could have different reactions and behaviors to the tasks and collaborations.

5.2 Results

The correct answer for the target location in the mystery is "NYSE" (New York Stock Exchange). On the one hand, no CI conditions found the exact correct answer. On the other hand, the uni-item and triple-item CI conditions found evidentially and geographically close locations, and provided meaningful analysis provenance that explains why the crowd made certain correct or wrong decisions. This good performance is consistent with prior evaluations of a similar pipeline [119]. In this section, we focus on reporting the types and

Steps	Uni-item GI	Uni-item CI	Triple-item CI	All-item CI	Step Total
1	45	45	15	3	63
2	20	18	8	2	48
3	57	48	15	3	123
4	15	24	9	3	51
5	2+8=10	2+8=10	2+8=10	2+8=10	8+32=40
Total	139+8=147	92+8=100	49+8=57	13+8=21	293+32=325

Table 5.1: Number of workers hired in each step and each condition, and the total number of workers in each step across conditions. While Step 5 only requires two workers (one writer and one reviewer), for the purposes of this chapter, we ran it 4 more times ($4\times2 = 8$ workers) to gather more data and be comparable to the other conditions.

sources of the errors crowds make in each step (RQ 3.1) and how crowd analysis in each step is different when given more local context (RQ 3.2).

5.2.1 RQ 3.1: Error Types and Propagation

We compare the crowd analysis under the GI uni-item and CI uni-item condition to investigate how the quality of input influences worker analysis performance. Counting the additional data collection in step 5, this includes 231+16=247 crowd workers in total: 139+8=147 workers for GI and 92+8=100 workers for CI (Table 5.1).

Different Numbers of Crowds Were Hired in GI and CI Conditions.

Passing on analysis results by previous crowd workers introduces uncertainty in the hiring process. Since the crowds selected 9 relevant documents in step 1, step 2 in CI condition only hired $9 \times 2 = 18$ workers to write and review info pieces. This resulted in fewer info pieces and thus, fewer workers were hired in CI step 3. In CI step 3, the crowd tagged additional irrelevant locations from the irrelevant information. Therefore, the CI step 4 hired more workers to evaluate the likelihood of all those locations.

Local Error Types and Examples

Comparing the crowd analysis with the gold standard analysis, we found local errors in each of steps 1–4. Crowd did well in step 5 with gold-standard input, and the errors in the CI condition are due to error propagation. We first categorize the local errors as follows, then discuss error propagation in the following subsection.

Insufficient context errors. In step 1, two relevant documents refer to terrorists by their aliases and were left out by the crowd. A different document reveals that those names are terrorist aliases. Insufficient context also led workers to rate irrelevant documents as relevant, because there was not enough information to prove the document irrelevant, and the information might be "worth looking into". In step 2, the information about terrorist aliases were not extracted. In step 3, the info pieces about terrorist aliases were tagged as not containing any relevant evidence. In step 4, many related locations are rated as likely target locations. One worker pointed out that "it's also very possible that it [Empire State Vending Services (ESVS)] is somewhere that they're just using as part of their cover stories", but still rated ESVS as a likely target.

Misinterpretation errors. In step 1, one relevant document was rated irrelevant by a worker. The explanation was "report date and deposit is dated after the date in question" but the dates in the document are actually before the attack date. In step 2, a worker extracted an info piece "Cedric Whappadder announced he would pick up the carpet..." that misinterprets the information in the document; Cedric Whappadder is the carpet store owner, rather than the customer. In step 3, a worker tagged "Sudan and Afghanistan" as one candidate location in info piece "Joed Shearper recieved explosive training Sudan and Afghanistan". This indicates that the worker did not understand that 1) Sudan and Afghanistan are two different

locations and 2) those locations are where the terrorists received training, not the targets of the attack. In step 4, a worker rated the New York Stock Exchange (NYSE) as irrelevant because "there is not a lot of mention about the stock exchange specifically". The worker only focused on the frequency of a location being mentioned, but did not interpret how this location is connected to the known facts of the attack.

Inattention to background knowledge. In step 1, one document directly mentioned a terrorist name, but workers rated it as irrelevant by the majority vote. Two of the workers did not mention anything about the terrorist name in the explanation. In another example, one irrelevant but misleading document about an attempted bombing was rated relevant, but it involves a different time from the known attack time. This indicates that some crowd workers did not pay attention to the known facts (e.g., terrorist names and the attack time) given in the instructions.

In step 2, the important information about the attack weapon (C-4 explosive) was not extracted. The workers only wrote about a cigarette being tossed into a waste basket in a carpet shop, but did not mention that this resulted in a fire and led the firemen to discover several cartons of C-4 explosives, nor did anyone mention that the carpet shop belongs to one of the terrorists.

In step 3, one crowd worker tagged an info piece, "Cedric Whappadder has C-4 explosives in the basement of his carpet shop until April 26, 2003" to have candidate location "in the basement of his carpet shop". This indicates that the worker did not pay attention to the known attack time (April 30) given in the instructions. The explosives are moved before the attack thus the carpet shop cannot be the target location.

In step 4, one worker explained that "it was written in the page above that bomb attack will take place in new jersey april 26 2003." This, too, conflicts with the known facts that the

attack was to take place on April 30, 2003. The worker might have mistaken the date for other dates mentioned in the task.

Failing the task goals. In step 1, a worker rated a relevant document as irrelevant, but pointed out the relevance of the document in the explanation: "Although the sentences describe how this attack may have been funded, there is nothing there that would make one aware of the location of the attack." The worker did not fulfill the task goal to rate documents as relevant if they contain information about the known facts of the terrorist attack.

In step 2, a worker extracted an info piece that reads, "I LISTED IN THE CITY NORTH BERGEN NJ ON APRIL 22,2003". This crowd worker put "I LISTED IN THE CITY" in the "what" field. This indicates that the worker did not follow the instructions to write complete sentences about the important information to solve the mystery.

In step 3, some workers put terrorist names as a candidate location tag. Some workers selected all the evidence tags, explaining, "I chose the tags above because it was stated in the instructions that they knew the weapon, the time and date, as well as, the group of terrorists who are expected to detonate the weapon." The worker did not understand that the task is to find the info pieces that are relevant to the known facts.

In step 4, a worker explained, "However there is no details related to the attack location or target of attack. It is extremely difficult to extract details." The worker didn't understand that the task is to rate the likelihood of the given location based on the available information in the profile.

Low effort errors. In step 1, one worker put "goode" in the explanation box. In step 2, several workers directly copied text from the documents to fill in the "who, what, where, when" fields that do not combine to read as a meaningful sentence. In step 3, some workers

put "Available Material" as a candidate location tag, and put "good" as the explanation. In step 4, some workers put "Available Material", or "this is clear" as the explanation.

Error Propagation

We analyze crowd error propagation by tracing the crowd analysis on previous errors (the left half of Fig. 5.2) and comparing it to the equivalent in the GI condition. The errors in earlier steps led to an increasing number of errors due to insufficient context and missing information (*inherited errors*), as well as more low effort and other local errors (*compounding errors*). On the other hand, some irrelevant information was filtered out in later steps (*accidental cure*). Below, we describe how each step is influenced by these types of error propagation.

Step 2 was influenced by inherited errors and compounding errors. Overall, step 2 only retrieved around half of the gold-standard info pieces. The CI condition missed all the information from the 3 missing relevant documents (*inherited errors*) and included additional irrelevant information from the 2 extra irrelevant documents (*compounding errors*). However, the CI condition extracted more matching info pieces than the GI condition, even though the input has fewer relevant documents. To further understand this surprising outcome, we analyzed the individual responses of the microtasks for the 7 relevant documents shared in both conditions. It turns out most errors (i.e., incomplete or irrelevant sentences) are due to local errors (*failing the task goals*). We speculate that the varied performance in the same task might be because the workers are overwhelmed or confused by the task and did not extract more info pieces than the minimum requirement. A follow-up experiment repeated step 2 for both the GI and CI conditions with the same microtasks but enforcing a minimum of 2 info pieces. The results confirmed this intuition. The new crowd ($N' = 20 + 18 = 38$) mostly extracted 2 info pieces, but the overall quality did not improve. Since the design choices are

consistent with the previous successful deployment of a similar pipeline pipeline [119], we suggest that extracting and restructuring information (step 2) is the most challenging step of the pipeline with more challenging dataset and longer documents.

Step 3 was influenced by all types of error propagation. The crowd-generated info pieces are less understandable due to incomplete and poorly structured sentences, typos and grammar errors, and some are written in all capital letters. As a consequence, some important information was tagged as not containing relevant evidence and introduced additional false negative errors (*inherited errors*). In addition, misleading information continued to be tagged by evidence types and propagated strongly. The crowd generated 8 location profiles, of which 3 are from irrelevant documents (*compounding errors*). There was also an increased number and percentage of meaningless explanations (*low effort errors*) in step 3. Most of them occurred in info pieces about aliases, phone numbers, etc., that require more context to tag. We suggest that the previous poor-quality analysis provided less context and might have confused the workers about the task goals. On the other hand, some false positive errors from step 1 and 2 was cured by step 3 crowd, because they couldn't find any evidence related to the known facts (*accidental cure*).

Step 4 was influenced by all types of error propagation. The CI condition crowd rated a fake apartment address of terrorists in NYC as the most likely target location. The correct answer NYSE was not in the step 4 input since the step 1 crowd did not rate the corresponding document as relevant (*inherited false negative error*). *The USA*, a very low-resolution location that nevertheless encompasses the correct answer, received almost the same score and ranked second place. Rating the apartment address as a likely target location, whose profile contains irrelevant and wrong information, as well as missing some important relevant information, is a *compounding error*. The irrelevant profile, on

the other hand, were rated as unlikely to be the target location (*accidental cure*). We further analyzed the explanations in CI step 4 to understand how the crowd was able to mitigate the previous errors. We found that the crowd compared the available information to the known facts, identified and excluded non-logical possibilities (e.g., locations not worth attacking), recognized cover-up activities (e.g., that the "carpet" to be picked up is actually C-4 explosives), and brought in their common sense for geographic proximity (e.g., identified explosive location in the carpet store in North Bergen, NJ and suggested the target is nearby).

Step 5 was influenced by all types of error propagation. While the GI presentation logically connects all 6 gold-standard facts (see Appendix B.1), the CI presentation only has 1 matched fact. The final CI presentation eliminated the two irrelevant pieces of information from the given location profile (*accidental cure*), revealed the connection between Cedric Whappadder with C-4 explosives, but reused the misinterpreted information (Cedric Whappadder picking up the carpet) from the wrong location profile (*inherited errors*), and connected the wrong information with the correct information (*compounding errors*). The additional data collected in step 5 is consistent with this result. The GI condition presentations are all complete and cohesive. Three of the new CI condition presentations inherited the wrong information about Cedric Whappadder picking up the carpet, one of which included more false positive errors from the wrong location profile. The other one new CI presentation is very short: "all terrorist attack attempt in April month". We consider this as a local error (low effort).

Overall, while the compounding of the errors in the CI condition is problematic, the GI condition of Study 1 suggests that if each step can be improved (perhaps through review or refinement processes), the chaining of the steps in a pipeline can be a successful strategy for crowd sensemaking.

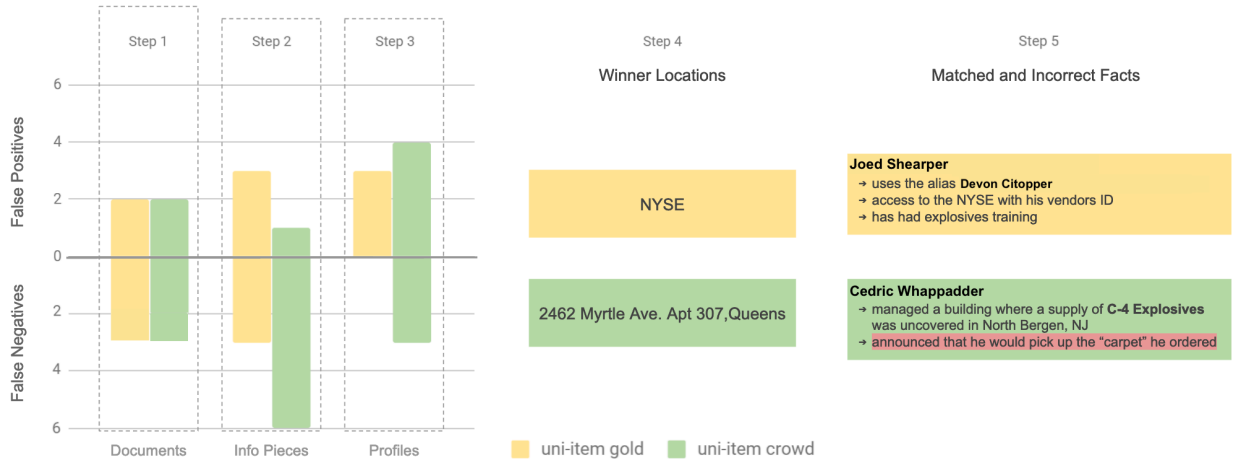

Figure 5.3: RQ 3.1: Errors in GI (gold-standard input) and CI (crowd-generated input) conditions.

5.2.2 RQ 3.2: Impact of Context

We analyze the crowd analysis in CI uni-item (Fig. B.1), triple-item (Fig. B.2), and all-item (Fig. B.3) conditions to investigate how the amount of local context influences the worker performance. Counting the additional data collection in step 5, this includes 154+24=178 crowd workers in total (Table 5.1).

Step 1

Increasing context changes frequency of different local errors. The uni-item condition retrieved 7 (out of 10) relevant documents and 2 (out of 5) irrelevant documents; the triple-item condition retrieved 9 relevant documents and 1 irrelevant document; the all-item condition retrieved all 10 relevant documents but also 4 irrelevant documents (Fig. 5.4).

Increasing context reduces unforced errors due to insufficient context. Both triple-item and all-item conditions successfully retrieved the one document about NYSE, while the uni-item condition failed to. Increased local context enabled workers to use in-

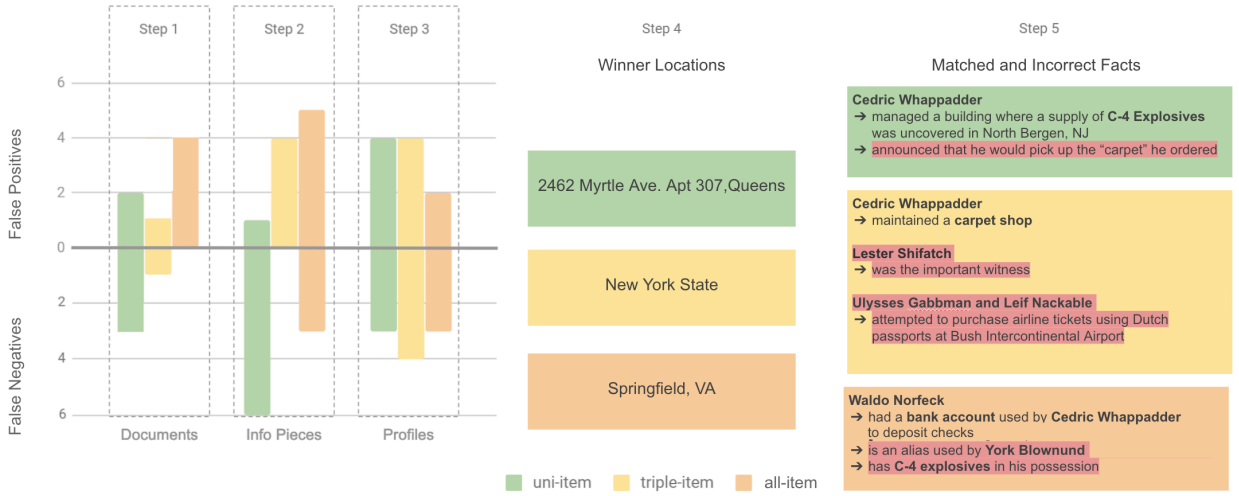

Figure 5.4: RQ 3.2: Errors in uni-item, triple-item and all-item conditions.

formation from different documents and retrieved documents with hidden relevance. The all-item condition saw the most references to related documents in the explanations.

Increasing context introduces unforced errors due to misleading context. On the other hand, the relevant documents in each context slice did not help crowds eliminate the irrelevant document. Rather, more irrelevant documents were rated as relevant. Our analysis of the explanations indicates that workers also drew connections between entities such as the country names and date time mentioned in the irrelevant and relevant documents (*unforced errors* due to misleading context). For example, one worker put in their explanation that "previous phone call made out from the Netherlands, and the passports are Dutch in this document." The call from the Netherlands is related to the terrorists in the mystery, but the Dutch passport is linked to a different crime.

Increasing context increases frequency of *inattention to the background knowledge.* In addition, workers in bigger context slice conditions are generally more likely to rate documents as relevant since they might contain "potential clues" in their explanations,

though we already listed the relevant terrorists in the instructions (*inattention to the background knowledge*). This might indicate that too much context prohibited workers from focusing on important information.

Increasing context increases frequency of *low effort* local errors. Compared to the 1 low effort explanation in uni-item (from 1/45 workers), there were 7 low effort explanations in triple-item (from 4/15 workers) and 30 low effort explanations in all-item (from 2/3 workers).

Step 2

Increased context overwhelmed workers. Workers in the uni-item condition extracted 16 info pieces from 9 documents (9 context slices), the triple-item condition extracted 12 info pieces from the 12 documents (3 context slices); all-item workers extracted 7 info pieces from the 14 documents (1 big context slice). None of the conditions extracted info pieces about NYSE.

Increased local context helped eliminate compounding false positive errors from step 1. The uni-item condition extracted info pieces from every document retrieved in step 1, thus some false positive errors propagated in step 2. The triple-item condition eliminated 1 of the 3 irrelevant documents. The all-item condition eliminated 2 of the 4 irrelevant documents.

Increased local context reduced *unforced errors.* The triple-item and all-item condition synthesized information from more than one document, which is not possible in uni-item condition. One crowd worker from the triple-item condition connected the phone number

addresses and the owner employment information. This reveals the identities involved in the mysterious phone calls reported in different documents.

Increased local context led to higher frequency of "failing the task goals" local errors. A higher workload led to more relevant documents being missed. In the uni-item condition, no relevant document was completely ignored, even though the information in the documents was not fully extracted. Yet, in the triple-item condition, 3 relevant documents were completely ignored, and in all-item condition, 5 relevant documents were completely ignored (*failing the task goals*). Fewer gold-standard info pieces and elements were retrieved in triple-item and all-item conditions, despite the increased availability of relevant documents. There were fewer relevant (although not gold-standard) info pieces and elements, as well.

Step 3

Increased local context reduced local errors, but also suffered more from propagated errors. The uni-item condition generated 8 profiles from the 16 info pieces, of which 3 are from irrelevant documents/info pieces. The triple-item condition generated 7 profiles from the 12 info pieces, of which 3 were from irrelevant documents/info pieces. The all-item condition generated 4 profiles from the 7 info pieces, of which 2 are from irrelevant documents/info pieces. None of the conditions created a profile for NYSE.

More context reduced unforced errors, but too much can lead to more low effort errors. In uni-item condition, 3 irrelevant info pieces were eliminated and 8 workers provided 8 low effort explanations. In triple-item condition, 1 irrelevant info piece was eliminated and 3 workers provided 5 low effort explanations. In all-item condition, 1 irrelevant info piece was eliminated and 2 workers provided 30 low effort explanations. We conjecture

that providing explanations to every single info piece encourages workers to analyze the info pieces more carefully, but could be too arduous with big context slices.

The distribution of context could limit accidental cures and encourage inherited errors. With our design, the context slices do not overlap, so it is hard to bring together the most optimal context without reusing the same info piece in multiple context slices. Some info pieces from step 2, though incomplete, could still make sense when put together with other info pieces. However, the KNN context slicing algorithm might not successfully put them in the same microtask, thus preventing effective tagging of those info pieces.

Step 4

Increased local context enhanced accidental cures with more in-depth analysis. Workers in the uni-item condition selected "2462 Myrtle Ave. Apt 307, Queens," the apartment address listed under two terrorist aliases, as the most likely target location. The triple-item condition selected "new york state" and the all-item condition selected "Springfield VA."

Increased context encouraged relative comparison and mitigated propagated errors. Despite the sparse information in each profile, the crowd was able to compare the relative likelihood of given locations with external knowledge and common sense. For example, one worker mentioned in the explanation that they recognized that "*the phone calls from Ramazi are originating from 703 area code — Virginia.*" Given insufficient information about each profile, some workers focused on eliminating unlikely profiles, rather than selecting likely ones. One explained, "*As for why I chose Springfield, it is the only unclear one. Two have no direct relation to the terrorist, and the third seems to be a home ad-*

dress." Although the final decision is farther from the correct location (NYSE), the analysis is more logical and accurate than workers who selected the apartment address of the terrorists. There were also zero low effort explanations in the triple-item and all-item conditions. We speculate this is because workers were less sure about their result and felt more obliged to explain their uncertainty and thought processes.

Step 5

Workers made more compounding errors. The winning profiles fed to step 5 in all conditions consist of almost half irrelevant info pieces. The workers introduced compounding errors by connecting the relevant info pieces with the irrelevant ones with dates since all documents are reports collected in mid-April 2003. The uni-item condition presentation contains 1 matched fact and 1 irrelevant fact. The triple-item condition presentation did not have any matched facts. There were 3 relevant facts connected to irrelevant information with dates. The presentation mostly described "evidence led the police to investigate…" The all-item condition presentation has 1 matched facts and 3 relevant facts. The two crowd workers build on top of each other's imagination and created a nice and long story with additional imagined information. Despite the false positive errors, the resulting prose is actually written in a similar fashion as the two-page long crime reconstruction given in the data set's answer sheet.

Crowds can also be distracted by recent news. Most of the presentations in the additional data collection are consistent with the original results. However, there are 3 presentations directly copied from recent news articles about the 2019 Sri Lanka Easter bombings. The original data was collected between February and March in 2019, but the additional data for step 5 was collected in June 2019. We suggest that the strong social

impact of the real-world attack could have distracted crowds from the analysis of a fictional mystery.

Overall, while crowd performance is negatively affected by too little or too much local context, the triple-item and all-item conditions of Study 2 suggest that the crowd can reliably synthesize distributed information and deduce hidden evidence, given the right amount and segmented local context, which varies based on the data and task design.

5.3 Discussion and Design Implications

In this chapter, we empirically investigated crowd errors and trade-offs of additional local context in different sensemaking stages. We categorized 5 major types of local errors, and inspected how they manifest in each step with different amounts of the local context. Below we first discuss how the error propagation in crowdsourced sensemaking resembles and differs from collaborative sensemaking among experts. We then draw the design recommendations for each sensemaking stage based on the crowd reaction to propagated errors and different amounts of local context. We also examine how the experiment setup could have influenced the crowd performance and the generalizability of our findings.

5.3.1 Error Propagation Among Crowds: Easier Hand-off but More Inaccuracy Blindness.

The pipeline clearly defines the step inputs and outputs, which makes it easier to distribute and aggregate crowd analysis. More importantly, it enables analyses of one step to be directly handed off to another. The step inputs and outputs serve as shared artifacts that facilitate crowd collaboration and eliminate errors due to misunderstanding and miscommunication.

This allows us to focus research efforts on the analytical errors.

To encourage volunteerism and avoid social loafing due to awareness of co-workers [99], we included workflow transparency in our task instructions. Workers were told that 1) the input they are given is from previous workers (except step 1 and the GI condition), and 2) their results will be used by future workers in later analysis. The workers were not told how many co-workers were working on the same part of the data. Despite the exposure to the pipeline workflow, however, the crowd still made low-effort errors. In addition, the crowd demonstrates strong team inaccuracy blindness [93]. Experts are cautious to re-use any given intermediate analysis and usually trace back to raw material to double check the if they agree with the given input [38]. In contrast, the majority of the crowd workers take the given analysis as correct, increasing inherited errors from previous steps.

5.3.2 Design Recommendations for Each Step

Searching and filtering tasks need more than one documents to better judge relevance, but smaller context slice sizes produce higher quality analysis in explanations (step 1). The optimal context slice size might differ by the investigation goals and the sizes of the data set. If the amount of raw materials is too big for experts to go through, the crowd can reliably handle 3–15 short documents (word count 1200–6000), if not more. 3 workers are enough to achieve reasonable accuracy via majority vote. On the other hand, if the experts aim to use crowds for more diverse perspectives with smaller data sets, assigning a smaller amount of data (about one document or word count 400) would encourage more thoughtful analysis that may be worth incorporating in expert analysis. In addition, when there is more than one document in each microtask, it might be helpful to require one explanation per microtask rather than per document, to balance the amount of

context and the workload. The disagreement in ratings is also an indicator of crowd uncertainty [150] and reveals the more difficult part of the data. If the raw materials contain data of multiple formats and granularity, it might be helpful to have a mixed design with some big slices and some small slices to balance the workload.

In terms of the task design, our pilot and actual studies demonstrated improved crowd analysis quality when setting an explicit threshold in a numerical rating task. For example, in our task design, the crowd was asked to give a rating from 0–100. We annotated the slider with 0 (completely irrelevant), 100 (completely relevant), with a box showing the selected value by the crowd and an indication of the value. (If it is above 50, the word "relevant" is shown next to the number, otherwise, it displays the word "irrelevant".) This helps to normalize the subjective rating preference of different individuals and supports more reliable result aggregation via majority vote.

Reading and extracting tasks might require further task decomposition and benefits from small, focused context slices (step 2). The current design of step 2 worked well in simpler data sets with shorter (word count 50–100) and easier documents [119], but cannot handle even one document written in report language (word count 400). For more difficult input data or when the crowd workers are not guaranteed to be native speakers, it is worth the effort to incorporate related research that focuses specifically on information extraction and hire more workers for each document. Successful examples include iterative re-representation [6] and the highlighting and clipping [82].

Schematizing tasks can handle big context slices and benefit from more effective hand-off with additional information (step 3). Restructuring information in the documents into simpler info pieces can support larger-scale information synthesis by increasing

analytical power and efficiency. When working with data of more concise formats, microtask performance benefits from bigger context slices. The crowd handled context slices of size 3 to 15 fairly well in our case studies, but the explanations became a burden. We expect the crowd can take even more than 15 data points per microtask, but it is also recommended to reduce the requirement on explanations, perhaps to one explanation per microtask, rather than per info piece. Similar to step 2, step 3 is yet another complex component that may benefit from being further modularized into sub-workflows. Schematizing is a more challenging step that connects the information foraging and synthesizing in the pipeline. It also challenges the local task with a global view more than other steps, since the organization of the information is required to make global sense and lead to further hypotheses. Successful related research could be incorporated in the pipeline to support this need and improve the task performance, such as using machine learning to pre-process info pieces and extract global patterns, and then focus crowd intelligence on edge cases [29], or having multiple iterations on crowd tagging results [35], or more effective sub-workflows and task interfaces [130].

Hypothesizing tasks benefit the most from bigger context slices to judge the relative likelihood and mitigate propagated errors (step 4). The crowd performance was not negatively affected by an increased number of profiles. Thus, we would expect the microtasks could handle 3-4 profiles (word count 1000), if not more. The main bottleneck of step 4 was that the correct answer was not even one of the available options. Besides improving the design of the previous step, we suggest the most effective refinement would be from a top-down path of the pipeline, with feedback provided based on a more global understanding of the data set and current analysis, to retrieve the missing information in previous steps and redo step 4.

Story-telling tasks might not benefit from additional context (step 5). In step 5, the crowd was given only one winner profile in all conditions, by design. However, the amount of information contained in the profiles ranged from 88 to 335 words. Thus, the crowd is able to handle at least this amount of information and write a reasonable report. When using the mixed-quality crowd analysis results as input, step 5 does not recover from errors and can inherit or even compound previous errors by connecting irrelevant information to the relevant evidence with coincidentally overlapping entities. In addition, our case study implemented a subtask in step 4 that allows workers to optionally merge profiles. This ended up providing richer information in step 5 that avoided losing precious relevant information and helped reveal the propagated errors. Thus, we also suggest that in similar pipelined crowdsourcing systems, the benefit of including more results from the previous step outweighs the potential disadvantages of introducing more false positive noise.

5.3.3 Generalizability and Future Work

Performance variance due to recruiting requirements and strategies. When the scale of collaboration increases, requesters must either spend more time recruiting more workers, or lower the recruiting requirements. Spending more time is not always possible, especially in time-constrained scenarios such as intelligence analysis. This chapter focuses on the problems caused by lowering the recruiting requirements. Our experiment results show that this leads to higher variance in crowd efforts and more low-effort performance. While the variance of crowd efforts is influenced by the sensemaking tasks and the context slice sizes, it does not differ by the input source.

Using time as a proxy measure of crowd effort, we found there is no significant difference in the crowd effort with crowd-generated and gold-standard input across all steps: $p_{step2} =$

$.079; p_{step3} = .98; p_{step4} = .65; p_{step5} = .26$. On the other hand, different sensemaking steps have different variances, and the variance is also influenced by the context slice size (Fig. 5.5). When using uni-item context slices, steps 2, 4, and 5 have higher variance than steps 1 and 3. Step 2 and 5 microtasks require free-text responses, which invokes more thinking and requires more time for some workers. Step 4 microtasks might be difficult to rate the likelihood depending on how much information the given candidate has. When using triple-item context slices, the difference in variance is smaller across the five steps, but step 5 is still the highest.

When using all-item context slices, step 1 and 3 took many crowd workers much more time and has a higher variance than other conditions. This might be because the microtasks for the two steps are more atomic (i.e., rating each document and tagging each information piece). Some workers might need to take breaks while finishing the microtask when the number of items increases. Step 2 and 5 variance is similar to other conditions, but the time required for all workers in step 2 is a lot higher than in other conditions, indicating that the microtasks get a lot more challenging for everyone. Step 4 takes a longer time but has a smaller variance, possibly because with all the candidates in sight, there is more to read, but picking one most likely candidate becomes easier.

Finally, microtasks for step 5 in all conditions are the same, and there is no statistically significant difference in the time spent ($p_{step5} = .31$). This result suggests that the crowd performance and errors can be generalized to a bigger pool of similar crowds.

One possible way to address the high variance and low effort problem is to raise the recruiting requirements. The extreme case would be expert collaboration [73]. Another strategy suitable for novice crowd workers would be to increase the number of workers per microtask. Our case studies used 3 workers for majority-vote tasks and 2 workers for create-review tasks. The results indicate that increasing the number of workers could help converge on

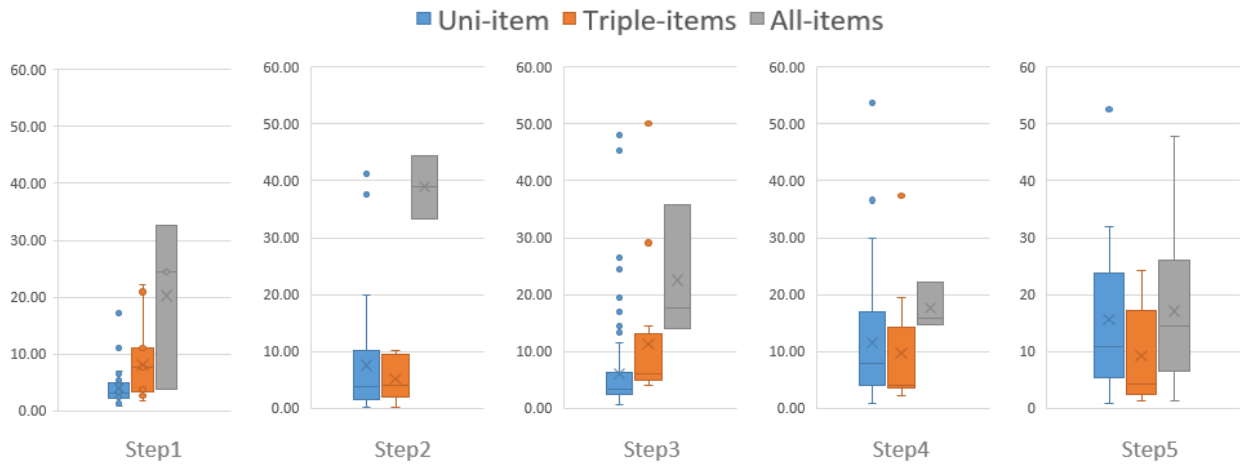

Figure 5.5: Time spent on each step when given different amount of local context

better analysis since poor performance is a smaller percentage than high-quality analysis. However, increasing the number of crowd workers might require changing the mechanism to aggregate crowd results. For example, in steps 2 and 5, an alternative would be hiring more crowd workers for the same context slice and then hire another group of workers to pick the best results. Future work is needed to investigate how the error frequency changes with different hiring requirements, and prioritize the design recommendations to eliminate the most influential errors.

Optimizing the analysis with task design and execution strategies. Besides our design recommendations to improve crowd analysis performance, additional subloops between connecting steps could also increase the quality of the intermediate step output. Our analysis of the explanations indicates that the crowd can sometimes recognize poor input and explain what is missing. This makes it promising to enable crowd-driven reporting of low-quality work (similar to the panic button proposed by Retelny et al. [155]) and redo the prior analysis. For example, if workers in a later step complain about the quality of input, they can be switched back to the previous step and fix the prior analysis. This forms

sub-loops between connecting steps before reaching the final step. Designers might need to put a limit on the number of iterations allowed between steps, to prevent long, inefficient sub-loops from wasting crowd workers' effort. Future work is needed to further quantify the crowd's ability to identify and self-correct analysis errors. Unlike rumors on social media [7], the task domain of mystery solving may not require the same amount of incentive and background knowledge to critique each other's analyses. Showing high-quality results from other workers is effective for making workers reconsider their judgments [110], and also brings in additional local context. However, this approach will make each worker stay for a longer session in each microtask, and require a mechanism to automatically pick the highest-quality results.

Top-down refinement with a global understanding of the data and crowd analysis. Given the unpredictable and challenging nature of exploratory sensemaking, we see the potential benefit of a top-down path of the pipeline to complement the bottom-up analysis. There is always a limit to how much context a local view of the data can synthesize, and even with a completely reasonable local analysis, the key evidence can be so well hidden that it cannot be revealed without iterative analysis. By the end of the pipeline execution, the crowd has produced a more detailed global understanding of the information available, which could help an expert prioritize and focus on the important evidence. Experts [54] or the crowd [133] could "debug" the pipeline and identify where mistakes were made during the sensemaking process. The pipeline structure provides built-in provenance analysis that traces the information and insights between steps, and makes it easy to examine the initial crowd analysis, evaluate analysis quality, identify information holes or logic flaws, trace back to errors in a top-down manner, and guide the refinement of the previous analyses with feedback. In future work, we plan to continue to explore how the pipeline can also support debugging and refining previous imperfect analysis. The errors and bottlenecks we

classified in this chapter can serve as a checklist to identify errors in previous crowd analysis. More importantly, the analysis provenance that connects the intermediate results and traces information flow will be critical for obtaining a big picture of the mystery and applying the newly acquired knowledge from the previous analysis in the refinement process.

Chapter 6

CrowdIA Pipeline: Top-down Refining Path

Sensemaking is "never fully complete" [38]. Crowdsourced sensemaking also needs to support the iterative refinement of the analysis. However, mixed-quality crowd results require significant efforts to curate and improve. Given the unpredictable and challenging nature of exploratory sensemaking, my work in Research Question 3 suggests the potential benefit of a top-down path of the pipeline to complement the bottom-up process. The errors and bottlenecks modeled in Research Question 3 can serve as a guideline for refining crowdsourced sensemaking. However, the crowd errors for a given analysis are specific to the data and problem, interdependent across multiple steps, and therefore nontrivial to identify and fix.

In this chapter, we explore **RQ4: How to refine the crowdsourced analysis with a top-down process?**

6.1 Preliminary Studies (RQ 4.1)

In this section, we report our exploration of **RQ 4.1: What are the key challenges in refining a pipeline of crowd analyses?** These pipelines have two key properties. First, they contain a pipeline of connected steps. The analyses are all crowdsourced and the results are passed from one step to the next as input. Second, the pipeline analyzes an

exploratory problem and uses text data. The problem is sufficiently complex but can be solved without prior knowledge or domain expertise. Recent examples from crowdsourcing research include extracting categories for unstructured text data (2 steps: re-represent and cluster) [5], researching purchase decisions (3 steps: map, partition, reduce) [101], or writing an article (5 steps: sourcing, clipping, clustering, integration, editing, multi-media) [80].

6.1.1 Can Crowds Refine Existing Analyses Directly?

We recruited 15 crowd workers on Amazon Mechanical Turk (MTurk) to directly refine an existing set of crowd analyses comprised of 5 steps. We assigned each step of the analysis to 3 workers, presented in a Google Doc. The crowds had a "suggestion only" view and the task was to improve the analysis by leaving comments or suggestions in the document.

The crowd left different types of comments, but none refined the given analysis. Some workers simply asked questions and many questions were unhelpful. For example, one worker asked "what weapons were used?", but this is one of the known clues in the mystery. Another worker asked, "where is the attack going to happen?", but answering this question is the overall goal of the analysis. Some comments were more specific about the analysis, but did not contribute to the current progress. One crowd worker commented that "this seems highly skeptical because, for most countries, money coming in from foreign bank accounts is monitored. Why would bank accounts in two different countries finance a grocery store?" The existing analyses already described money laundering activities to fund an explosive attack. This comment highlights one correct part of the existing analyses, but does not improve the quality. Furthermore, the crowd was not able to distinguish strong versus weak (or even wrong) hypotheses. For example, one crowd worker commented that "even though this person had the means, motive and opportunity, there's no evidence of that person

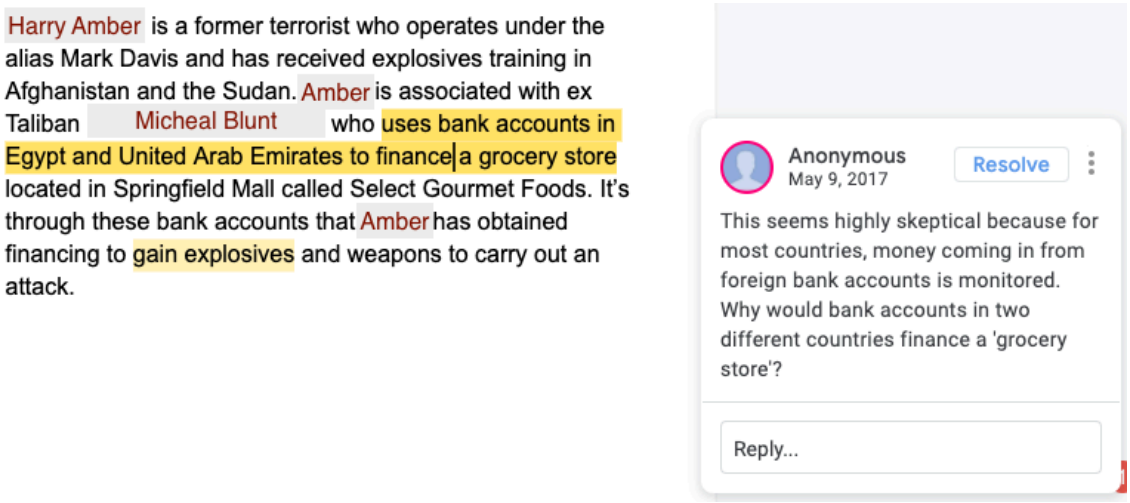

Figure 6.1: Example of crowd comments in the first preliminary study.

[having] actually killed the victim." This is a reasonable comment, but a key challenge in exploratory analysis is to establish hypotheses from incomplete information and evaluate their relative strengths. In other cases, crowds questioned the accuracy of the known facts rather than the analysis. One worker commented, "how do I know if information in the document is true or designed to confuse me?" More importantly, the crowds did not suggest any refinement of the given analysis.

The results of the first preliminary study indicate that crowd workers with only a partial view of the analyses cannot refine different parts of the analyses independently, or provide feedback that helps fix those problems.

6.1.2 Can Crowds Fix Identified Problems?

Assuming the problems are already identified, can the crowds help fix those problems? We designed possible mistakes in different steps of crowd analyses and constructed feedback for each problem in the format of *context, critique, and to-do*. *Context* defines the part of the data and corresponding analyses that has a problem. *Critique* explains why this is a problem.

To-do proposes how this problem could be solved. We created a microtask for each problem and recruited 3 crowd workers on MTurk for each microtask. A total of 48 crowd workers participated in this study.

Most crowd workers were able to correct the mistakes we designed in different steps. Taking the majority vote eliminated most of the poor results in the refining tasks. Specifically, the crowds were good at correcting low-effort errors (e.g., incorrect document transcriptions), as well as verifying if some analyses contained irrelevant noise (i.e., false-positive errors). When a mistake was inherited from previous steps, the crowds were also able to suggest that the mistake cannot be corrected within the current step. In addition, we found that the crowd performed better when given specific action items. For example, "extract the information about the motive of Joe to kill the victim from document X" produced better results than a high-level description of the problem like "this suspect does not have the motive to kill the victim".

Thus, we found that crowd workers can reliably fix problems, assuming that the problems have been identified with structured and specific feedback.

6.1.3 Can Individuals Identify Problems in Crowd Analysis?

Crowds are limited by local context to effectively identify problems. Can individuals with a global view of the analysis do better? We recruited 6 lab participants to identify problems with the crowd analyses. Each session lasted one hour.

Participants identified several important problems, but needed support to overcome information overload. Most participants spent more than 30 minutes (i.e., most of the session) to make sense of the task. They were overwhelmed by the complexity and the amount of information: the background of the problem, the structure and content of crowd analyses,

Error Location		Critique	Error Type	Todo	
Column	*Item ID*	*(Why this is an error)*	*(Categorize by reasons of errors)*	*What input will the crowd need to fix the error?*	*What will be your instructions?*
Crowd Info Pieces	74	Crowd worker provided a unhelpful/incomplete analysis	Low effort (Meaningless analysis)	document 2	Include more relevant details mentioned in document 2 (date, time, location)
Crowd Hypotheses	NA	Carl Louis' arrest date/year was not mentioned in the final hypothesis. This might be vital information.	-- Propagated Error (The error originates from earlier steps)	israel synogogue in detroit MI, 2462 myrtle ave apt	Include relevant dates in the hypothesis
Crowd Info Pieces	72	It is wrongly stated that Carl Louis was arrested on April 27, but that was actually the date of the report.	-- Propagated Error (The error originates from earlier steps)	document 15	Read the document carefully
Crowd Info Pieces	71	It is wrongly stated that Carl Louis was arrested on April 2, but the document doesn't actually state when he was arrested	-- Local Error (The error started in this step)	document 15	Read the document carefully
Crowd Documents	15	abhelhak kherbane's arrest in connection with the synogogue bombing is not captured	-- Local Error (The error started in this step)	document 15	Read the document carefully

Figure 6.2: Problem description table for participants to fill in and example results by participants.

and how the crowd collaborated to produce the existing analyses. This left little time for identifying problems. While all participants found several problems with the crowd analyses, they described the problems in an abstract and high-level way. For example, one participant said she "didn't find a clear mapping between hypotheses and presentation" and identified this as a problem, but she didn't specify what she meant by a clear mapping.

Participants provided feedback to fix the problems identified, but needed support to specify more actionable details for crowds. When asked to provide feedback for crowd workers, one participant wrote, "look for more information about this person". However, to refine the analyses, crowd workers require more specific and actionable feedback to guide them in finding the missing information.

6.1.4 Design Goals: Supporting Crowd Auditing

As we conducted the preliminary studies, we began to see the emerging role of the committed analyst as a kind of *auditor*. In the business world, auditors are external analysts with two key responsibilities: finding problems within an organization, and proposing solutions. Taking inspiration from this model, we conceptualize the analyst's goal as *crowd auditing*.

External perspective. In financial audits, the auditor is external to the organization. This unique, outside perspective provides two key advantages. First, the auditor is not a member of the organization, so they bring a fresh set of eyes and a more neutral perspective. Second, unlike most employees, the auditor has a global view of the organization's data and processes. Our preliminary studies suggested a separation between analyst and crowd was similarly advantageous. *A crowd auditing system should preserve this complementary division of labor and its affordances.*

Finding problems. A key method for finding problems in business auditing is through a systematic review of financial records, tracking provenance and following the evidence trails. Our preliminary studies found that crowd workers struggled to find problems. Individual analysts with a global view were more successful, but they experienced information overload. *A crowd auditing system should help auditors balance the global context and information overload so they can identify problems.*

Proposing solutions. Financial auditors provide feedback on an organization's process and suggest potential local fixes as well as broader process improvements. However, the auditor doesn't implement the solutions; it is the client organization's job to act on those suggestions. Our preliminary studies found that crowd workers could not directly refine existing analyses, but performed well when given actionable feedback. Individual analysts, however, struggled to provide clear instructions. *A crowd auditing system should scaffold auditors to provide specific, actionable feedback to enable crowds to solve problems with their analysis.*

In the following section, we describe CrowdTrace, a software tool we developed that demonstrates the crowd auditing model based on these three design goals.

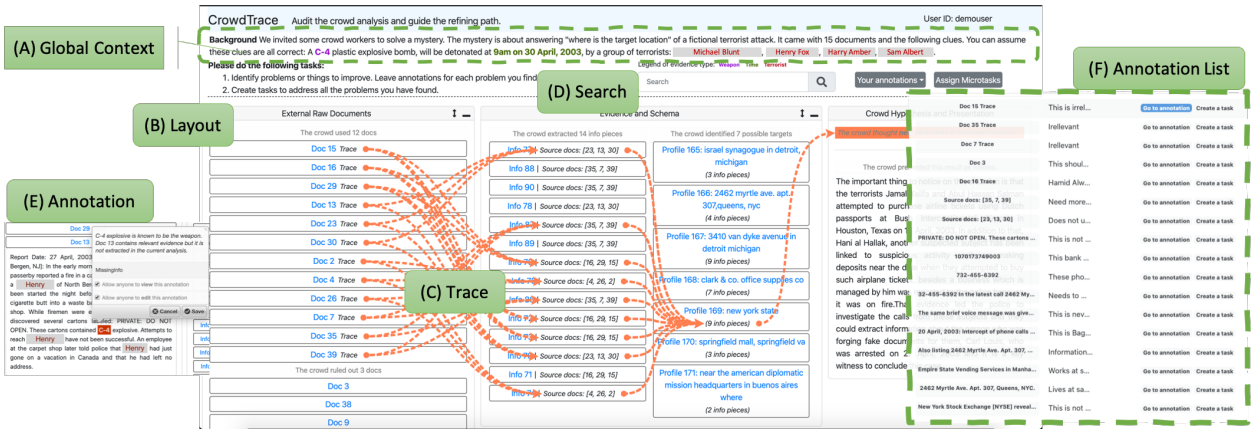

Figure 6.3: CrowdTrace. Auditing interface for identifying problems,

6.2 CrowdTrace (RQ 4.2)

In this section, we present CrowdTrace, to address **RQ 4.2: How can technology be designed to support refinement of a pipelined crowd analysis?** CrowdTrace is implemented with the Django web framework and deployed on the Heroku cloud platform. The back end is written in Python with a PostgreSQL database and uses the boto3 API to communicate with MTurk. The front end is implemented with the Bootstrap UI framework in HTML, CSS, and JavaScript / JQuery.

6.2.1 Overview

CrowdTrace imports a *global context* (problem background and description) and the existing crowdsourced analyses with a specification of 1) the number of *steps* (N), 2) data *format* (in HTML) of each step, and 3) the crowd work *history* (3.a. mapping from the input data in each step to each microtask, 3.b. mapping between the crowd results and each microtask).

CrowdTrace supports two sub-tasks in crowd auditing: identify problems and provide feedback. This separation of concerns is drawn on the observation from pilot studies, where

participants split the auditing task into these two parts naturally.

CrowdTrace presents the global context of the problem at the top of the screen (Figure 6.3 A) in both sub-tasks. Below the global context, CrowdTrace displays the crowd analysis. The crowd analyses are laid out in the order of provenance in different columns (B). The first column is the raw dataset, and the last column is the final presentation of the crowd analysis. Middle columns are the intermediate step outputs. Each column has multiple data items, such as documents, information pieces, and candidate answers. Clicking on the item titles in each column can expand or collapse the corresponding content. The microtask creation interface is hidden by default and can be accessed with the "assign a microtask" button.

6.2.2 Identifying Problems

Trace the Analysis Provenance Auditors can hover over the "trace" button next to each item title to show the source information used to create the current item, and the downstream analyses (Figure 6.3 C). The provenance is displayed with arrows that point from the source items to the current item, as well as from the current item to the downstream items. Each arrow only connects items in adjacent columns. The auditor can also lock the flow of provenance by clicking on the trace button, which will keep all the items in the information flow highlighted.

We provided these two tracing mechanisms based on the participants' feedback from the pilot studies. Participants had trouble understanding "what do these analyses mean" as well as "what did each crowd worker do". They wanted to trace "where this [piece of analysis] came from" and "where did it go", with "arrows" showing "the direction of information". Participants also expressed that they want to "lock" each thread of analysis, so that they can investigate the items and their relationships with other items in the analysis.

Search for Keywords and Highlight Threads of Evidence CrowdTrace allows auditors to search for occurrences of different words and phrases (D). All matched occurrences are displayed by expanding the corresponding items. Other items without any occurrence will be collapsed accordingly. Also, keywords in the global context are highlighted by categories of evidence. CrowdTrace supports easy search of keywords in the global context by clicking on them directly.

The implementation of the search feature is tailored for crowd auditing based on the requirements expressed by participants in the pilot study. Participants wanted to examine "where is this keyword in each step", to see "where this information got lost". With the crowd analyses displayed in order of provenance, coupled with the tracing features, auditors can examine the keywords occurrences in each step, and compare the analysis about the same keyword in different locations.

Annotate on Problems and Take Notes Auditors can highlight any part of the existing analysis and leave an annotation. Each annotation contains highlighted text, comments, and optional tags (E). All annotations created by the user can be easily accessed and retrieved through a drop-down list on the upper right (F). Users can click on "go to annotation" to locate the context and details of an annotation, or "create a task" to fix the problem described in an annotation.

We implemented the annotation features to fulfill the requirements from the participants to "take notes" and "leave comments" in different parts of the crowd analyses. Participants needed to take notes in context of the existing analyses. With the in-place annotation feature, the auditors can highlight the problematic parts and describe the problem in the local context. Participants also wanted to review and retrieve their annotations more easily. With the annotation list and the "go to annotation button", auditors can review their auditing

outcome, and refer back to the local context of each annotation in different places of the crowd analysis.

6.2.3 Providing Feedback

Provide Feedback by Creating Microtasks CrowdTrace supports auditors to provide feedback by directly formulating the feedback as microtasks. This design decision is based on a pilot study where participants structure their feedback by filling in a template table on a Google Sheet (Figure 6.2). The resulting error descriptions were much more detailed than in previous pilot studies, but are still not specific and clear enough to be transformed into microtasks. We decided to ask auditors to directly create microtasks instead of writing feedback that needs to be transformed into microtasks later. To support novice auditors in creating microtasks, we provide a template in a simulated preview of the microtask interface.

Simulated Microtask UI with a Template When the auditor creates a microtask, CrowdTrace displays a preview of the microtask interface that crowd workers will see (Figure 6.4). The preview includes the background and the goal to refine the existing analysis. Auditors need to fill in 4 blanks in the template, 1) describe the problem, 2) provide instructions for the crowds to fix the problem, 3) provide the corresponding input information for the crowds to work on, 4) specify the requirements of the format of the crowd answers. All 4 blanks need to be completed to create a microtask.

All-in-one-place View to Mitigate Context Switch Overhead Auditors can create a microtask from an annotation (F), or can click on the "Assign Microtasks" button on the upper right of the interface to create a microtask from an empty template. When opened, the microtask creation interface is stacked on top of the crowd analyses. If the auditor creates a

Figure 6.4: Microtask creation interface and example microtask created by a participant.

microtask from an annotation, CrowdTrace automatically imports the annotation comments as the problem description, and the highlighted part of the existing analysis as the input information. Furthermore, CrowdTrace supports directly importing input information from the existing crowd analysis. Each column of the existing analysis has a separate import button in the microtask creation interface, laid out in the same order as in the analysis provenance.

We designed this all-in-one-place according to the pilot study results where the interface for providing feedback was in a separate window. Providing feedback in a separate window caused extra overhead due to context switch. Participants repeatedly forgot their thoughts before filling out a row in the sheet; one commented, "I wish these two [crowd results and google sheet] can be in the same view." Creating microtasks in the same view also allows auditors to refer back to different parts of the crowd analysis and their annotations more easily.

6.2.4 Example User Scenario

Jamie self identifies as proficient in English reading comprehension. She is hired to audit a crowd analysis of a mystery. Jamie decides to start her auditing by reading the *Crowd Hypothesis* and *Presentation*. She notices that some known clues are not mentioned. Jamie searches for one of the missing clues, *C-4*, As a result, Doc 13 is expanded with the occurrences of C-4 are highlighted. This means C-4 is mentioned in Doc 13 but no other places in the analysis. Jamie hovers on the trace button in Doc 13. It shows that 3 downstream information pieces are related to Doc 13. She locks this flow and looks at each info piece. It seems like the crowds overlooked this part of Doc 13 about *C-4*. She highlights the sentences about *C-4* in Doc 13 and leaves an annotation to identify this problem.

As the number of identified problems gets larger Jamie wants to review her auditing progress. She navigates to her annotation list by clicking on "Your annotations". She clicks on "go to annotation" in her first annotation. Doc 13 is expanded and she sees her highlighting and comments. Reminded of the problem about C-4, she clicks the "create a task" button in this annotation. Jamie now sees the microtask creation interface stacked on top of the crowd analysis. The template instructions provide her with a sense of what the crowd workers will see. Her comments and highlighted text in the annotation are already imported in the template. She quickly rephrases her already detailed problem description and writes a bulleted list of task instructions (1. Read Doc 13, 2. Extract the information about the *C-4* clue mentioned in the document). Then, she imports Doc 13 in the input information using the "Import Docs" button. Jamie specifies the format requirement to be "a short list of detailed sentences" and then clicks on "Save changes". Steadily, Jamie continues her task creation by going through the list of annotations.

6.3 Evaluation Study Design

We conducted a user study to evaluate our CrowdTrace and inform the design of future crowd auditing tools. Our evaluation is guided by two primary criteria:

C1 **Performance:** How well does CrowdTrace support crowd auditing? Specifically, how well does CrowdTrace support 1) identifying problems and 2) creating microtasks to fix the problems?

C2 **Efficiency and Learnability**: How much time is needed to conduct each part of crowd auditing, and how well do auditors learn to use CrowdTrace?

6.3.1 Performance Metrics

We evaluate participants' performance by 1) comparing the problems identified to a gold standard list of important problems, and 2) recruiting crowd workers to evaluate the quality of the microtasks created by each participant.

Developing a Gold-standard List of Important Problems Two of the authors compared the existing crowd analysis to the solution of the mystery. The dataset solution lists important facts and evidence needed to solve the mystery. First, the authors identified *important problems* as those containing three attributes: 1) an error description (why this is an error), 2) a local error location (where this problem originated from), and 3) the propagated error locations (where and how this error influenced the later analysis). Each of the two authors developed a list of important problems separately. Second, the two authors consolidated their lists of important problems through an in-depth discussion. The problems are itemized to support a more granular analysis. For example, the existing crowd analysis missed information about the fake home address and employment of a terrorist (Henry) under an alias (Joe). This is broken down into 3 problems: a) missing information about Joe being an alias of a terrorist called Henry, b) missing information about Joe's home address, and c) missing information about Joe's employment with Company X. Propagated errors are considered to be the same problem as the original error, and not listed as a separate problem. In total, we identified 26 important problems with the existing crowd analyses.

Measuring Quality of Microtask with Crowds To measure the quality of the auditors' microtasks, we employed a task design and bonus policy inspired by [136]. We invited crowd workers to work on the microtasks created by the auditor participants (Part I) and rate the quality of the microtasks (Part II). Part II has two questions. Question a) asks workers to rate

the clarity of the auditor's problem description, task instruction, and format requirement, as well as the sufficiency of input information. Question b) asks the crowd if they found confusions with the task definition. If yes, the crowd can ask a clarification question and provide their best guess to answer the question. If no, the crowd needs to briefly explain their understanding of the microtask. The crowd can also optionally provide additional feedback. Crowds are bonused for completing Part II according to their performance. The bonus policy for each question in Part II was listed in a table next to the question.

6.3.2 Participants

We posted study recruitment advertisements in the classrooms of a large research university. People who signed up needed to fill in a screening survey. We selected participants who self-identified their proficiency level in English reading comprehension as intermediate or above, and do not have prior experience as a crowd worker or requester. Twenty-four people signed up for the study, and all passed the screening survey.

Due to the recent COVID situation, all study sessions had to be conducted remotely via video call. We updated the IRB consent form and 5 people decided to drop out. As a result, the user study had 19 participants to serve as auditors. The participants ranged in age from 18–29, 5 were female and 14 were male.

6.3.3 Procedure

Before each session, the participants were asked to fill out a pre-survey to collect their demographic information. During each session, we first introduced the background of the study and gave a tutorial of the CrowdTrace system. After that, the participants were given an hour to complete the two subtasks in any order. The participants were asked to think

aloud during their audit. In the end, the participants were asked to fill out a post-survey to share their experience working on the crowd auditing tasks.

After the lab studies, we recruited crowd workers on Amazon Mechanical Turk to work on the microtasks created by the auditors and evaluate their quality. We recruited U.S. crowd workers who had completed at least 100 HITs (Human Intelligence Tasks) with an approval rate of at least 95%. For each microtask created by an auditor, we recruited 3 crowd workers to work on the microtask (a HIT) and evaluated the task quality for bonuses. Each HIT was worth $0.84 to complete Part I, based on a wage of $7.25 / hour and average pilot deployment durations of 7 minutes. Two researchers evaluated the crowd results for Part II independently (inter-rater agreement k=0.73, indicating sufficient agreement), and resolved the disagreements together before sending out bonuses.

We also conducted qualitative analyses on the annotations. Two of the authors separately compared the annotations created by each participant and coded the problems identified. If the problem matched the list of important problems, the corresponding ID of that problem was noted.

6.4 Results

The auditor participants each left an average of 14 annotations in the crowd analysis (min=6, max=31, median=12) and created an average of 6 microtasks (min=2, max=10, median=6) (Figure 6.5). The crowd workers spent an average of 6.5 minutes on each microtask and received a total payment of $0.96 on average. In this section, we focus on reporting the performance of crowd auditing (RQ 4.1), and the efficiency and learnability of the CrowdTrace system (RQ 4.2).

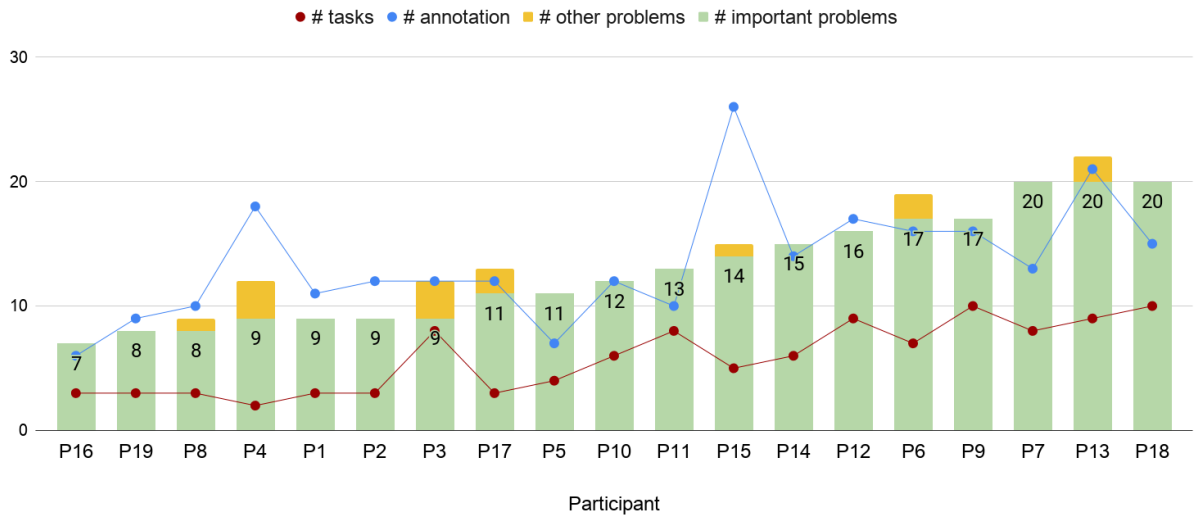

Figure 6.5: The stacked bars show the number of important and other problems identified by each participant, sorted by the number of important problems; the lines show the number of annotations and microtasks created by each participant.

6.4.1 Performance of Crowd Auditing

The two authors compared the coding and consolidated the differences in the qualitative analysis of annotations (inter-rater agreement k=0.82, i.e. very good agreement). In addition to the important problems, we also found annotations that describe "other problems" (if the comments describe legitimate problems with the crowd analysis but not listed in the golden standard list, for example, typos or grammar errors); "auditor mistakes" (if the comments contain a mistake, for example, considering a correct analysis as wrong); or "note to self" (if the comments do not identify a problem or contain any mistakes).

Participants focused on identifying the important problems. Overall, each participant identified an average of 13 out of the 26 important problems (min=6, max=21, median=12). Most of the annotations were used to identify one or more problems with the crowd analysis, with a few identifying "other problems" and "note to self". The annota-

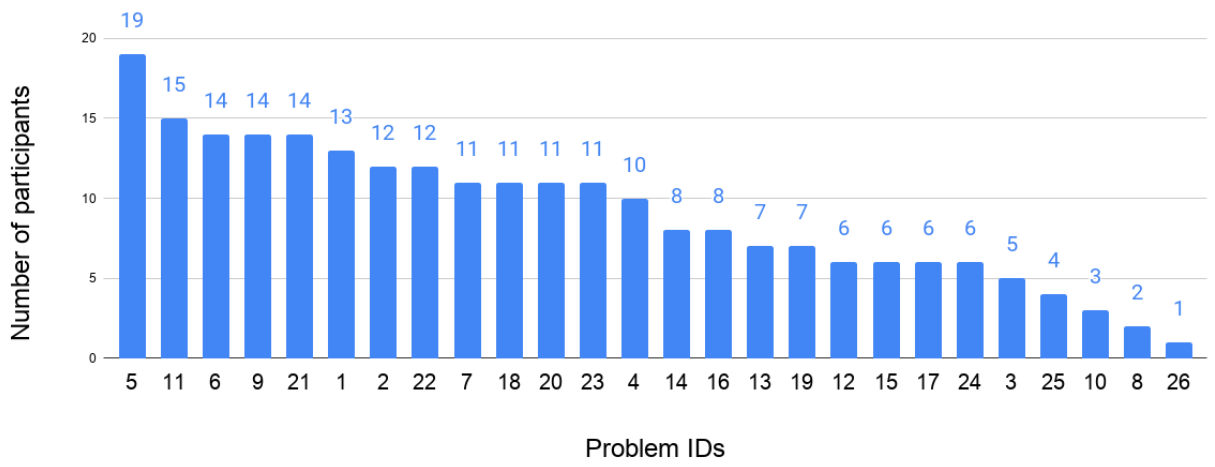

Figure 6.6: The number of participants who successfully identified each problem, sorted by frequency.

tions that describe "other problems" were mostly created in the earlier stage of each session. We only found 4 auditor mistakes with 2 participants. Figure 6.5 shows that the number of annotations created is close to the number of problems identified by most participants (N=15). For the other participants, P4 and P15 used many annotations to take notes of the evidence discovered from the dataset. P7 and P18 left more detailed annotations that typically describe more than one problem with the analysis.

Some problems were more successfully identified. All the important problems were identified by at least one participant (Figure 6.6). Meanwhile, some problems were more successfully identified than others. For example, all 19 participants identified the problem with an alias of a terrorist (Henry) being missing (Problem ID=5). Other more successfully identified problems are also related to this thread of evidence: the missing information about Henry's explosive training (Problem ID=6), missing information about C-4 explosives (Problem ID=11), missing information about the address of Henry (Problem ID=9) and that another terrorist also lives in the same address (Problem ID=21). On the other hand, only 1 participant identified that extracted information about a terrorist was not included in

the final presentation (Problem ID=26). Less successfully identified problems were mostly local errors in the later steps (Problem ID=26,25,24), irrelevant information included in the existing analysis (Problem ID=10,3), or less conspicuous information hidden in a long document with other relevant information (Problem ID=8).

6.4.2 Performance of Creating Microtasks

The participants created 110 microtasks in total on CrowdTrace. As a result, we created 110 HITs on MTurk to evaluate each microtask, recruiting a total of 330 crowd workers. 6 of the workers turned out to be bots; we rejected their work and recruited additional workers to complete the work.

Crowd workers found microtasks clear, actionable. 254 (77%) workers rated the problem specification in the given microtask as "very clear" or "clear". 246 (76%) workers rated the task specification in the given microtask as "very clear" or "clear". 244 (74%) workers rated the input information in the given microtask as "very sufficient" or "sufficient". Finally, 237 (72%) workers rated the format requirement in the given microtask as "very clear" or "clear".

When asked if found any confusions with the given microtask, 223 (67.6%) of the crowd workers did not find any confusions. Two of the authors analyzed the clarifying questions asked by the remaining 107 crowd workers who found confusions (inter-rater agreement k=0.76, i.e. sufficient agreement). 56 out of the 107 workers did not ask meaningful questions. Examples of such questions include writing a comment instead of question: "there are some minor confusions but overall it's good" or repeating the instruction: "what does 'a list of sentences' mean? It means several sentences in a list." Among the 51 crowd workers who asked meaningful questions, 18 were able to provide legitimate best guesses.

In summary, workers rated 93 (84.5%) microtasks created by the participants as high quality, i.e., clear and providing sufficient input. 6 (0.018%) microtasks were rated lower quality but still understandable and doable by crowds. 11 (0.03%) of the microtasks had unclear instructions or insufficient input.

6.4.3 Audit Strategies

Two of the authors conducted a content analysis of the video and audio recordings of each participant's audit process. The two authors each analyzed half of the recordings and compared the patterns observed (Figure 6.7).

Auditing Direction We identified 4 reoccurring directional patterns of the overall audit process. In the **bottom-up** pattern, an auditor starts the audit by reading through the documents and work towards later steps. The auditor primarily focuses on the raw data instead of the crowd analysis, similar to a *linear* process of developing conclusions from the raw data in intelligence analysis [94]. In the **top-down** pattern, an auditor starts by reading the final presentation and work backward to earlier steps. This adds to the linear process and focuses on testing the crowd-generated hypotheses against the raw data [109]. In the **browsing through clues** pattern, an auditor searches for the keywords in the known clues and look for the occurrences. In the **following the leads** pattern, an auditor works through each thread of evidence across different steps. Browsing through clues and following the leads reflect the "parallelism" in intelligence analysis, where different steps are examined almost at the same time, but the amount of time spent focused on each step changes throughout the course of auditing [94].

Participants tended to employ a combination of the above patterns or move from one pattern to another. Browsing through clues and following the leads were usually used together. Par-

Category	Subtype		Template
Direction	Sequential	Bottom-Up or Top-Down	Participant X starts analysis by reading (all documents | crowd presentation)
	Nonsequential	Keyword Search	Participant X searches keywords by clicking through the known clues in the global context
		Lead-Following	Participant X follows a lead [person | date | location] by digging deeper in [N] thread(s) of information
Step Focus			During the analysis process, there was a heavy focus on [N] step(s): (Step 1 | Step 2 | Step 3/4 | Step 5)
Establishing Relevance			Relevance of any information is established by (relating it to the global context | frequency of information in the dataset)
Annotation & Task Creation	In Alternation or in Subsequence		Creating [one | all] annotation(s) [first | per step | per lead] then [one | all] task(s) [per step | per lead]
Annotation Style	Detailed or Brief		[Most] annotations are (self-notes | identifying problems) with (overly brief description without context | detailed reasoning connecting different information)

Figure 6.7: A typology of audit strategies used and externalized by participants in crowd auditing.

ticipants who were able to identify more important problems mostly used the combination of these two patterns. We hypothesize that this is due to the participants grounding themselves in known and verified information and creating a stronger association between the analysis by focusing on one lead at a time. The linear processes, on the other hand, tended to cause some observed information overload.

Establishing Relevance We also found that participants who found more important problems established the relevance of information by relating it to the known clues of the mystery. For example, P18 left an annotation saying that "[this document is an] irrelevant document, as it does not mention names/alias[es] linked to the group of terrorists". On the other hand, we also observed a poor strategy where some participants established the relevance of information by the frequency of that information. For example, P5 searched the keyword "Carl Louis" (an irrelevant name included in the existing analysis) and found multiple occurrences across the five steps in the existing analysis. "This name appeared in so many places, maybe it is relevant." P5 did not identify that "Carl Louis" is an irrelevant person that was mistakenly included in the analysis.

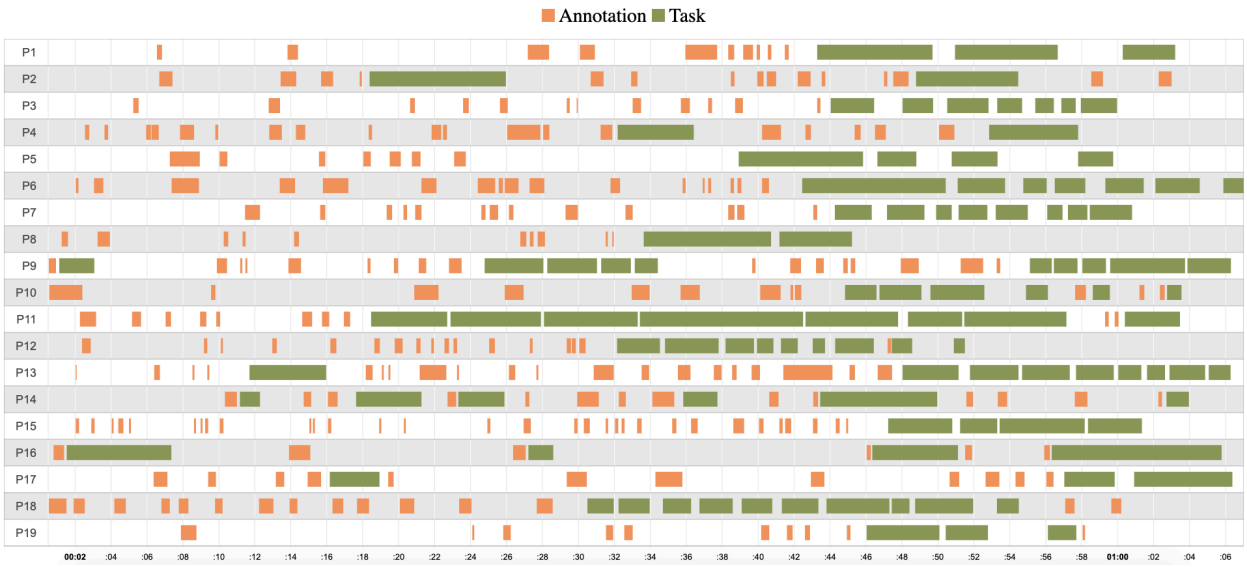

Figure 6.8: Time management of each auditor. Orange bars are time spent on creating annotations, and olive green bars are time spent on creating microtasks.

Time Management Between the Two Sub-tasks Participants also arranged their time and effort differently to complete the two sub-tasks. Figure 6.8 shows the timeline for the two sub-tasks by each participant. Overall, many participants first identified problems then created microtasks. Some participants worked on the two sub-tasks in parallel or switched between the two by threads of clues. Some participants identified new problems while creating microtasks (when they looked back after annotating multiple problems). There was no strong correlation between time management and the performance of participants. However, we observed some key trade-offs. If a participant creates microtasks while identifying problems, the original flow of thoughts is interrupted. The participant would forget more of the previous insights and spend extra time repeating previously identified problems. On the other hand, if a participant spends the first half of the time on identifying problems and the second half on creating microtasks, they would spend more time reminding themself of the previous annotations, therefore spending more time overall on creating each task.

Annotation Styles The time spent on reminding oneself of previous insights also was related to the annotation styles. Some participants left detailed comments in each annotation. For example, one of the annotations created by P18 was "This is a relevant document because this shows the numbers that the terrorists were communicating from and shows the alias and locations of them." When creating a microtask, P18 was able to quickly remember the context of each problem, observably provide a detailed problem definition, and create microtasks with relatively less amount of time (ranging from 1 to 4 minutes) compared to other participants. On the other hand, P12's annotations were all very simple without much context information, such as "[this is] not [an] important person involved in analysis" or "more information to gather on this alias." P12 only created 3 microtasks, spending 4 to 7 minutes on each. We hypothesize that combining a detailed annotation style with the subsequent completion of the two sub-tasks could lead to fast context recall and microtask creation for the participants along with more overall information exposure which would help refine the scope of the microtask.

6.4.4 Efficiency and Learnability: Time Measurement

Easy to Get Started We use the time elapsed before starting to create the first annotation to measure the time needed to get started in crowd auditing. The time ranged from 56 seconds to 12.3 minutes (median=2.8 minutes, mean=4.7 minutes, sd=3.4). Comparing to the results in preliminary studies, where the participants spent more than 30 minutes to start identifying problems, CrowdTrace enabled participants to quickly get started with the crowd auditing task within a few minutes.

Identifying Important Problems Gets Faster Over Time We use the time elapsed before creating each annotation, whether the previous operation was creating an annotation

Figure 6.9: Left: Time elapsed before annotation (minutes) decreases as more annotations are made. Right: Time spent on creating each microtask decreases as more microtasks are created.

or a microtask, to measure the time spent on identifying problems. The participants spent an average of 2.3 minutes (median=1.5 min, sd=2.55) to identify a problem. Fitting the time spent on identifying problems and the number of problems in a linear model (Figure 6.9 left) indicates that the time spent on identifying a problem significantly decreases as the participants identify more problems (estimate=-0.13, $p<0.001$). The average number of important problems identified in each annotation does not correlate to the timestamp at which an annotation is made. In other words, the participants got faster at identifying important problems.

Furthermore, the variance of time spent on identifying a new problem by different participants also decrease over time (estimate=-0.15, $p<0.001$). This indicates that even if a participant starts off identifying problems slowly at the beginning of the session, s/he can reliably learn and use 2-3 minutes to identifies a problem.

Creating Microtasks Gets Faster Overtime When creating microtasks, participants needed to refer back to their annotations and existing crowd analyses to remind themselves of their discoveries and import input information for the microtasks. Overall, participants spent an average of 2.34 minutes on creating a microtask (median=2.9 g2 min, sd=1.99). Fitting the time spent on creating microtasks and the number of microtasks in a linear model (Figure 6.9 Right) indicates that the time spent on creating a microtask significantly decreases as the participants create more microtasks (estimate=-0.32, $p<0.001$).

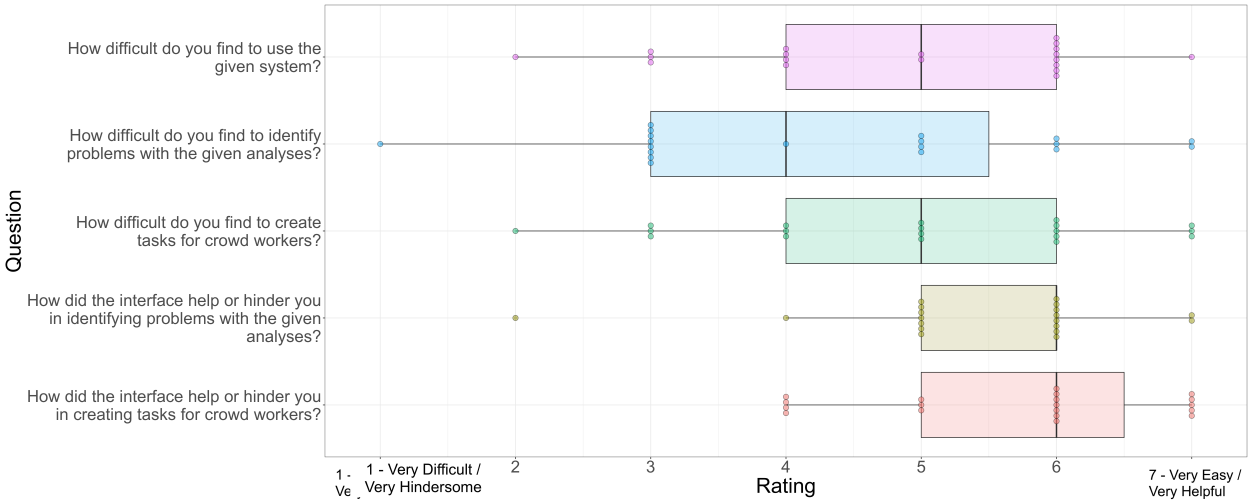

Figure 6.10: Questions and responses in the post survey. Individual responses are shown in dots.

6.4.5 Efficiency and Learnability: Survey Response

CrowdTrace Considered Helpful for Crowd Auditing Tasks In the post-survey, we asked the participants about their experience working on the auditing tasks and using CrowdTrace (Figure 6.10). The difficulty of crowd auditing tasks is perceived differently by participants. But almost all participants (89.5%) found CrowdTrace helpful for crowd auditing tasks.

The participants were also asked to share what they liked and disliked about their experience auditing crowd analysis. Many participants (N=11) mentioned that being able to **trace** the provenance helps them understand the given analysis. P4 said that tracing "helped me easily visualize where multiple information piece was coming from in regards to a person analysis." P2 found that tracing "makes it easy for the user to read stuff and focus on the relevant documents." P15 found tracing helpful for "seeing where the transition of information broke down and writing a task on it." P13 indicated that "it [tracing] made it easy on the eyes and mind to follow."

Some participants (N=7) mentioned that keyword **search** was favorable: "I liked how if I wanted to find something I could search for it and it would show me where it was found in every document" (P18). Participants also liked the microtask creation interface: "Creating tasks was very easy to use, especially once the annotation was made" (P13). The simulated view of microtasks helped participants to communicate their intention to crowds; for example, P12 wrote, "I also liked that you were shown exactly what the crowdsourced individual [crowd workers] would see when creating a [micro]task, so you knew what information to assume they did and didn't know." Some participants mentioned that they liked that all auditing tasks can be completed in **one view**: "very nice to have all the information in one place and did not have to go across multiple tabs or wait for things to load" (P11).

Challenges for Crowd Auditing with CrowdTrace When asked what they did not like about the crowd auditing experience, 4 participants did not report anything. Some participants (N=7) complained about lagging issues, but acknowledged that it might be "probably a hardware issue on my part" (P6) or "zoom doing video encoding in the background" (P13). Some participants suggested additional features that would help them, such as "a place to take general notes other then annotations" (P1), or "I wish there was a 'previous search' button [or] if I could flag the documents I had open, that would be better" (P14). Many participants also commented on information overload (N=6), e.g., "it was hard to remember which document contained the information" (P9).

Crowd Analysis Considered Helpful for Problem-solving Participants considered crowd analysis helpful for problem-solving. "I liked the fact that there was some analysis done and I got to work backward from the final presentation of the target back into the raw documents" (P3). 15 (78.9%) participants selected "yes" when asked if they would prefer to have crowd analysis available when solving a similar problem. The step-by-step structure of

the existing analysis was also considered helpful to understand and audit: “all the required information was presented in an easily digestible format” (P16).

6.5 Discussion

In this work, we investigated how to support iterative refinement in complex crowdsourced sensemaking. We proposed the concept of crowd auditing and developed CrowdTrace, a system to facilitate novice auditors to identify problems and create microtasks to refine a pipeline of crowd analyses. In this section, we discuss the challenges and opportunities in crowd auditing, our lessons learned about scaffolding crowd auditing, and future work in crowd auditing in a broader context.

6.5.1 Challenges and opportunities in Crowd Auditing

Crowd auditing differs from prior crowdsourcing research in quality control as it does not only focus on evaluating crowd performance and improving the design of a single microtask, but also on probing the problems in an existing pipeline of crowd results and steering the refinement with a global context. Below we detail the new challenges and trade-offs in crowd auditing.

Information overload vs. reference point Auditing a given set of crowd analyses requires the auditor to evaluate more information than solving the problem from scratch. The auditor needs to understand the processes in which the crowd generated the existing analyses, the distribution of information, and the corresponding analysis provenance. On the other hand, having existing analyses available eliminates the cognitive burden to digest

and synthesize unfamiliar raw data into meaningful structures and hypotheses. The crowd analysis offers a starting point for auditors and domain experts to quickly understand the raw data and strategically manage their efforts to focus on the more important parts of the analysis. In addition, the crowd results can serve as a reference point for the analysts, to compare their interpretation of the data with the crowd opinions [141]. This helps relieve the cognitive load due to the uncertainty of exploratory analysis, as well as enhancing the overall analysis outcome.

Identifying problems, provide feedback, fix the problems Crowd auditing requires a global view of the data and the analyses to effectively identify problems and direct the refinement effort to fix problems at the source. However, crowds with local views of the analyses are limited to effectively refine existing analyses. On the other hand, refining specific parts of the existing analysis in practice are formulaic and do not require a global context, thus is suitable to be completed with microtasks. It is hard to critique the existing analysis and provide feedback at the same time. These are two different tasks that require different levels of thinking, but are related since the person who critiques a part of the analysis is in the best position to provide feedback to improve the problematic parts. Handing this off to a different analyst will cause extra cognitive burden to understand other people's critique [199]. As a result, we divide the two as subtasks in crowd auditing processes and support iteration between them.

Specific feedback incurs extra workload vs. effective hand-off Furthermore, the more specific and detailed the critiques and feedback are, the more effort is needed from the auditor, and the closer the auditor is to directly solve the problem. However, crowds add value by actually working on improving the existing analysis with the feedback. It requires non-trivial efforts to act on the feedback, for example, extracting the missing information

in a needed format. Furthermore, writing specific critiques and actionable microtasks helps auditors remember their work better and provide more context to remind themselves in the future, which will be beneficial for future iterations.

Crowdsourcing novices as auditor vs. additional support Choosing who to be auditors is yet another challenge. In this work, we recruited university students who have no prior experience with crowdsourcing to audit crowd analysis. The goal is to get generalizable results applicable to different domain experts. Lacking experience and skills to create microtasks incurs communication barriers between the auditors and the crowd workers. Designing a microtask creation interface for crowdsourcing novices plays a crucial role to involve domains experts without prior crowdsourcing experience to harness the wisdom of crowds in various sensemaking processes. We will discuss more on this in the next subsection.

6.5.2 Design Implications for Scaffolding Crowd Auditing

In this section, we discuss lessons learned about scaffolding crowd auditing with CrowdTrace and draw design implications for future research.

Context-Aware Provenance Analysis Identifying problems in crowd analysis requires an understanding of the original local context seen by the crowd workers who produced the local analysis, as well as an understanding of the missing context needed to fix the problems.

Due diligence: awareness of global context. In CrowdTrace, we found that having always-visible known clues in the interface and highlighting the keywords in both the clues and documents raise auditors' awareness of the overall analysis goal and help keep them focused. Auditors who identified more important problems tend to concentrate more on the known clues and follow the leads.

This principle echos the due diligence auditing process in business contexts [83]. The goal is to evaluate the "climate" of the business and establish the objectives and postulates to plan and scope the later auditing effort.

Analytical procedures: awareness of local context. Crowdsourced analyses are usually constructed from local analyses of different "context slices" [122]. Understanding the available local context for previous crowds is important to identify problems and provide feedback. A crowdsourcing novice may wonder "why is this information not used, while being available?" However, crowds with local context sometimes do not recognize the relevance of the information. Visually tracing the original context slices and the corresponding local analyses help novice auditors understand the crowd analysis provenance. Furthermore, tracing the analysis provenance also enables auditors to focus on one thread of clues at a time and mitigate information overload.

This principle echos auditing techniques such as inspection and inquiry in financial auditing [69]. The goal is to establish an understanding of the specific client processes, evaluate the information available, and probe relationships among the data.

Re-slice the context for refinement. Refining the analysis would require creating new context slices to compensate for the missing context in the previous distribution of crowd work. Keyword search with auto expand/collapse enables auditors to organize the data and analyses into new context slices to support the effective refinement of the current analyses. The existing crowd analysis could also be used in new context slices to provide a broader context without introducing extraneous workload. For example, if information from 5 different documents needs to be connected to develop hypotheses, and the information has already been extracted and formulated in the form of simple sentences in the previous analysis, the refining microtask can use the more condensed sentences instead of requiring crowd workers to read 5 longer documents to draw the connections.

This principle echos the selection and sampling techniques [63] for obtaining audit evidence and drawing implications for audit reports in financial accounting.

Microtask Creation as a Communication Artifact Providing actionable and specific feedback for crowds is challenging for both expert and novice requesters [20, 136]. In crowd auditing, we found it effective to frame the providing feedback as creating microtasks to scaffolding communication between auditors and crowds.

Simulated Preview of Microtask Interface. Our preliminary study results and iterative design process revealed that auditors need a mechanism to help them cognitively switch from "talking to themselves" to "talking to a crowd worker" when creating a microtask. In CrowdTrace, we draw on prior research for supporting novice requesters [79, 136] and developed a streamlined microtask creation interface for crowd auditing. CrowdTrace displays the "what you see is what you get" preview of the microtask interface and helps auditors understand "what information to assume they [crowd workers] did and didn't know" (P12). This helps auditors communicate their intentions and expectations to crowds with more specific instructions.

Import annotations and previous analysis. In CrowdTrace, auditors can create a microtask by filling in the blanks or by importing their work from annotations. Reusing annotation comments helps auditors remind themselves of their previous discoveries. This mitigates the context switch between identifying problems and creating microtasks. Auditors can also import items in the existing analysis directly into a microtask. This connects the existing analysis with the microtasks that are designed to refine the analysis, keeps the refinement work focused on the problematic part of the previous analysis, and more importantly preserves the structure of the crowd analysis to enable future iterations.

Easy retrieval and management of annotations. CrowdTrace lists all the annotations created by each participant at the upper right corner of the interface. Auditors can easily look

up for annotations and go to each annotation in the corresponding locations. An overview of all the annotations helps auditors keep track of their auditing progress. The "go to annotation" button supports quick retrieval of the annotations and reminds auditors of previous discoveries. Participants also used this feature to optimize their microtask creation. For example, P2 identified missing information about a terrorist's alias and his explosive training background. These are two problems but he only created one microtask, since both pieces of information are contained in one document. He merged the two microtasks into one to optimize the crowd effort.

6.5.3 Generalizability and Broader Impacts

In complex and non-homogeneous processes, such as generating clusters from unstructured data [5], constructing an article to answer a given question [80] or solving a mystery [120], multiple steps of crowd analysis are involved and the quality of work is difficult to control. Errors and mistakes in intermediate steps would also impact the later analysis and overall outcome. Therefore, although the best practices in task and workflow design can help improve the quality of work in each intermediate step, researchers and practitioners still need mechanisms to refine a pipeline of imperfect crowd results.

Crowd auditing for iterative crowdsourced analysis The quality of crowdsourced analysis is one of the major bottlenecks in the application of crowdsourcing [136]. Typical crowdsoursable tasks such as labeling training data or looking up for a piece of information can have its quality of work improved by smarter or iterative task design, as well as inviting more people to work on the same task. Yet the crowd results are still not perfect. Crowd auditing addresses the challenge of quality control from a different perspective from prior research, embracing the imperfection of human work and focusing on supporting iterative

refinement. Crowd analyses have been shown to augment individual productivity and individual analysts can also optimize crowd effort in the analysis [141]. Refining the multi-step crowd process introduces additional challenges of identifying and tracing the propagation of each problem, sorting out the relationships within compounding errors, and prioritizing the sequence in which to fix each problem. In this work, we demonstrate one working mechanism that uses auditor–crowd collaboration to address this complexity of crowd auditing. Crowd auditing might also be suitable for enhancing the quality of crowd results in other complex sensemaking processes that involve multiple steps, such as extracting categories for unstructured text data [5, 33], trip planning [195], online shopping [105], or researching home improvement solutions [80]. Future work is needed to support longer-term iterations, such as deciding whether and which new crowd results are good enough to be included in the analysis, and when to use the new results to execute later steps again and generate further analysis.

This work utilized novice auditors in crowd auditing. CrowdTrace provides a mechanism for asynchronous collaboration between the auditor and crowds. This eliminates the time commitment of auditors and crowds to stay online at the same time during the analysis, and allows more flexibility in terms of who is the auditor (a committed crowd worker, domain experts, etc.). Future research can further explore different types of auditors, such as domain experts or even crowd workers. Another open question is when the number of auditors is more than one, how to aggregate and distribute the resulting microtasks. One possible way is to organize the microtasks by the corresponding input information, and add another layer of crowd work to merge similar microtasks.

More intellectually stimulating microtasks for novice crowds Effective crowd auditing can not only improve the quality of crowdsourced sensemaking outcomes, but also

creates opportunities for engaging more diverse crowds in a broader variety of problems. On modern crowdsourcing platforms, novice crowds are mostly swamped with simple, repetitive microtasks such as data labeling and transcription. Continuous long hours working on such tasks can risk the physical and mental health of crowd workers [77]. Both auditors and crowd workers voluntarily reported that the mystery-solving task is "interesting" and "fun". Exposing novice crowds to more intellectually stimulating microtasks can help improve crowd work and market places, as well as induce more diverse opinions and leverage broader "wisdom of crowds" in modern sensemaking challenges.

Mixed-initiative systems and AI transparency Another possible application is to use the auditing methods to help interpret decisions made by mixed-initiative systems. Modern sensemaking processes now involve more data and agents, such as crowds, domain experts, and AI. Interpreting the system decision-making process is important for fairness, such as algorithm-driven hiring processes; expert intervention, such as explaining mixed-initiative decisions in supply chain management; content auditing, such as identifying misinformation in online discourse. Quality control in these processes is difficult due to the additional complexity of multiple agents. Future work is needed to adapt the system to accommodate different schemas in sensemaking processes or to induce structure in unstructured processes.

Chapter 7

Conclusion

In this chapter, I conclude my dissertation and discuss future work. I begin by returning to the challenges and goals posted in Chapter 1, showing how the research presented in my dissertation addresses each of the research questions. I then summarize the major contributions offered by this dissertation. Finally, I discuss opportunities for future work in decentralized sensemaking.

7.1 Addressing the Research Questions

For this dissertation, I posed four research questions focusing on challenges and goals for crowdsourced sensemaking.

7.1.1 RQ 1: How to Enable Crowds to Extract Hidden Connections and Produce a Holistic View of the Information in Text Documents with Local Views of the Data?

The first challenge is to achieve a holistic view of the data with distributed local contributions. This is also a prerequisite for uncovering the hidden connections between entities to reveal the hidden plots in the data. One main difficulty is that there are no clear definitions of what is relevant or important. I address this challenge by modularizing the data with the

concept of "context slices". Context slices decompose the dataset into smaller but contextualized subsets to enable in-depth inquiry by transient novice crowd workers on Amazon Mechanical Turk. In this research question, I focused on the problem of supporting expert intelligence analysis using text datasets containing various documents. Specifically, I explored how to crowdsource the critical step of extracting the hidden connections between entities from the documents. In this case, context slices are smaller groups of documents. The challenge for crowds is to find the important connections for the entire analysis while only having a local view of the data (a context slice). The connections should also be compatible with each other so that the crowd results in each context slice can be aggregated into a global view of the relationship network.

To explore the utility of context slices, I developed a web application, *Connect the Dots*. The system automatically assigns an incoming crowd worker with a context slice of the data that needs to be analyzed, records and aggregates the crowd results for each context slice. *Connect the Dots* renders a crowd task interface that presents the text documents in a context slice and visualizes the entities as dots in a canvas. The crowds can select two dots from the documents or the canvas to make a connection. Each connection requires a brief description of the relationship between the two entities.

Using *Connect the Dots*, I conducted an experiment in which crowds generated connections using context slices of a text dataset. The dataset was borrowed from a course for training intelligence analysts. We compared the crowd results to the gold standard solution for the dataset that was generated by the course instructor. The experiment results showed that when their results are aggregated, crowd workers connected 92% of entity pairs mentioned in the gold standard solution. The node degrees, often indicating node importance, in crowd-generated graphs are very similar to those in the graph built from the solution. The results show that entity pairs that are connected in the solution are also more likely to be connected

by more crowd workers. I identified three types of connections that the crowds generate, and characterized the challenges of avoiding or excluding less useful types. I also found that double-document slices with overlapping entities outperform other slicing methods. Besides, a majority vote with a threshold can effectively improve the precision and recall values of crowd-generated connections. Furthermore, qualitative analysis of the crowd-generated connection descriptions (edge labels) shows that the crowd-generated descriptions for the connections appear to converge well and provide accurate and understandable summaries of entity relationships.

With Research Question 1, I explored non-expert crowds' potential to support a complex sensemaking process of expert analysts with context slices. The results indicate that context slices can assist novice crowds in eliciting key information from text documents and construct a global view of the dataset. With "context slices", the crowds can work in parallel and independently on easy and small tasks to find almost all entities pairs in their given contexts that were mentioned in the solution. With a reasonable threshold (4 votes out of all 5 crowd workers), we can achieve high values of both precision and recall. Context slices composed of documents with overlapping entities lead to better analysis quality. Furthermore, I identified three types of crowd connections: contextual connections, common-sense connections, and collateral connections. The insight can guide future work to automatically categorize and guide crowds in making more connections of the needed types.

This work brought the crowds into the expert sensemaking loop. The crowd demonstrates promising performance to make meaningful and useful contributions to expert processes. Connections needed by expert analysts in their investigation process are more likely to receive higher votes from workers. Moreover, rankings of node degrees from the crowd-built graph are similar to those from the gold standard graph. This indicates that crowds have the potential to pre-process the raw data and provide a reliable reference for experts by

prioritizing the entities. Taken together, these results indicate the efficacy of context slices to support crowdsourced sensemaking.

7.1.2 RQ 2: How to Establish a Bottom-up Pipeline to Enable Many Distributed Agents to Develop a Theory from the Raw Data?

Research Question 1 explored how to extract and discover the key relationships between entities with crowds. Schematizing key evidence is at a middle stage in the entire sensemaking process. Although the crowds were able to outperform machine learning algorithms and generate a meaningful network of entity relationships, I realized that the large crowd-generated relationship network is difficult to reuse and understand by a different group of crowds. Furthermore, the input data was assumed to be perfectly relevant documents and the named entities are pre-categorized, both require non-trivial work from previous sensemaking steps.

To make this work more practical and applicable to a broader domain of problems, we need to connect crowd work in different sensemaking tasks with minimal expert intervention. This brings us to the second challenge: enabling large-scale collaboration on highly integrated cognitive activities. The sensemaking loop [154] contains multiple inter-connected sub-loops. The data needs to be transformed in its raw format through different sensemaking steps into the final presentation of the analysis results. I address this challenge by modularizing the process into a sensemaking pipeline and expanding the concept of context slices to modularize the data in each step.

Starting from the copious external data source, we first need to construct a bottom-up building path of the pipeline to develop a theory from the raw data. Based on the sensemaking loop proposed by Pirolli and Card [153], an empirical study of the expert sensemaking pro-

cess [39], and research on crowdsourcing workflows and task designs for sensemaking tasks [11, 81, 95, 102, 177], I modularized the bottom-up sensemaking process into a pipeline of five steps defined by the analysis inputs and outputs, where the output is passed to the next step as input. Each step applies context slicing to enable crowdsourced analysis. I implemented the pipeline as CrowdIA, a web-based crowdsourcing system that provides automated facilitation of the sensemaking process for novice transient crowd workers. The system takes a set of raw documents as raw input, and automatically create context slices to support crowd collaboration. After all crowd workers submitted their results, CrowdIA aggregates the context slice analyses into step output, and pass that to the next step until reaching the end of the pipeline. The outcome is a presentation of the analysis results in the format of a narrative paragraph.

With CrowdIA, crowd workers on MTurk successfully solved mysteries of different difficulty levels. Context slices in each step enabled meaningful and scalable division of work. Crowd workers in later steps were able to recover from many false-positive mistakes from earlier steps without expert intervention. For example, in the difficult dataset, the Step 1 crowd included one irrelevant document (a false positive), which propagated the useless information to later steps. Nonetheless, Step 3 guaranteed that only useful information pieces are tagged with evidence types and included in profiles of candidates. Thus, useless information was filtered out in Step 3. Furthermore, the crowd was also able to provide explanations for their local analysis, which contained diverse perspectives. On the other hand, the crowd unavoidably made various mistakes. However, the pipeline naturally preserves the crowd analysis provenance, which enables step-wise debugging of their sensemaking process. Meanwhile, the modularization of the process enhanced the scalability and reusability of the crowd analysis. For example, future work can explore alternative strategies to schematize information (step 3) without redoing the other steps. Future research can also explore possible ways

to optimize each step with best-suited technologies, as well as human-AI teaming within and across steps. In addition, context slices in each step facilitated an efficient division of labor. The pipeline also introduces future research opportunities to explore alternative workflows to execute the steps. The pipeline enables research on different sensemaking steps to be dynamically plugged in and tested, thereby coordinating large-scale efforts from the sensemaking research community.

We chose to deploy our pipeline to solve intelligence analysis mysteries because they exemplify the challenge of exploratory analysis. However, we envision the pipeline as adaptable to broader applications with different sensemaking challenges, as well as opening up more in-depth research within each step. Furthermore, the pipeline can support flexible crowd compositions and collaboration settings. Our evaluation study deployed CrowdIA on MTurk, which is a paid online crowdsourcing market place composed of a general public crowd. When confidentiality is a concern, in addition to incorporating task assignment techniques for sensitive documents, it is also possible to use a trusted internal group, or a more specific group of experts on other platforms, such as Upwork. Overall, the hope is that this pipeline will serve to accelerate research on sensemaking, and contribute to helping people conduct in-depth investigations of large collections of information.

7.1.3 RQ 3: What are the Limitations and Challenges in the Bottom-up Pipeline for Crowd Sensemaking?

The results from Research Question 2 suggests that crowds do not always succeed when collaborating on sensemaking tasks without expert intervention. Although crowdsourced sensemaking has demonstrated impressive potential in a range of complex tasks and domains, mixed-quality crowd work remains a key challenge for crowdsourcing research. As

we learned from Research Question 2, chaining multiple crowdsourcing processes without expert intervention could cause the crowd errors in each step to propagate and compound in later steps. This is the third challenge this dissertation aims to address.

Using the CrowdIA pipeline as a testbed, I conducted a series of mixed-method studies with 325 crowd workers to probe the errors and bottlenecks in crowdsourced sensemaking. I first investigated how crowd performance is influenced when given either crowd-generated input or gold-standard input (RQ3.1). I then examined how the amount of local data context influences worker performance and error propagation (RQ3.2). The number of crowd workers hired in each step is dependent on the analysis results in the previous step. As a result, the number of crowd workers recruited to analyze a data set can be different every time the pipeline is executed. However, the analysis outcome by the pipeline is stable: we executed the pipeline to analyze the same mystery 5 times on MTurk under the same conditions. The different crowds all reached the same final answer, even though the intermediate analyses varied.

I evaluated the crowd performance by comparing their analysis to a gold-standard analysis adapted from the dataset's answer sheet. I classified 5 types of *local errors* that occurred in each intermediate step, specifically focusing on the source and impact of errors: 1) insufficient context, 2) misinterpretation, 3) inattention to background knowledge, 4) failing the task goals, and 5) low effort. The results also showed how chaining together mixed-quality crowd analyses can inherit or even compound previous errors. I analyzed how the errors in each step propagated to later steps, as well as how each step is influenced by the errors from previous steps. For example, wrongly retrieved irrelevant documents or useless information pieces can further pollute later analysis. Surprisingly, both false positive and false negative errors were mitigated to some extent without external mediation. I attribute this to the design of the pipeline that condenses information from documents to info pieces to profiles

as the analysis progresses to high-level goals.

I modeled the crowd errors in a two-dimensional space defined by the *input data* and the *crowd behavior*. When provided with correct input data and doing the task right, the crowd would achieve *ground truth analysis*. However, if they don't do the task right, they will make *local errors*. When the local errors are passed to the next steps, even if the later crowd workers do the task right, chances are the errors will be *inherited*. Meanwhile, there is also a possibility that the later crowd ignores and excludes the wrong parts from the analysis. Finally, if the errors are passed to the later steps and the new crowds don't do the task right, there will be *compounding errors*.

The results also demonstrated the impact of different amounts of the local context. Increasing the amount of local context impacts the crowd errors differently in different steps. Generally, a more local context can facilitate synthesizing information from distributed sources but would introduce additional workload that can overwhelm workers and increase the number of local errors. I also found that the pipeline achieved an easier hand-off of analysis among a big number of crowd workers but suffered from more inaccuracy blindness. Based on the above analysis, I proposed design recommendations for supporting complex crowdsourced sensemaking, both within individual steps and across the broader pipeline. I also provided insights into how crowd performance can vary depending on the recruiting requirements and strategies.

7.1.4 RQ 4: How to Refine the Crowdsourced Analysis with a Top-down Process?

Mixed-quality crowd results require significant efforts to curate and improve. Given the unpredictable and challenging nature of exploratory sensemaking, my work in Research Ques-

tion 3 suggests the potential benefit of a top-down path of the pipeline to complement the bottom-up process. The errors and bottlenecks modeled in Research Question 3 can serve as a guideline for refining crowdsourced sensemaking. However, the crowd errors for a given analysis are specific to the data and problem, interdependent across multiple steps, and therefore nontrivial to identify and fix.

We first explore the design space and investigate the main challenges in refining mixed-quality results in a pipeline of crowd analyses. As we conducted the preliminary studies, we began to see the emerging role of the committed analyst as a kind of auditor. In the business world, auditors are external analysts with two key responsibilities: finding problems within an organization and proposing solutions. Taking inspiration from this model, we conceptualize the analyst's goal as *crowd auditing*. Our preliminary studies suggested a separation between analyst and crowd can provide a fresh set of eyes and a more neutral perspective to examine the crowd analysis. Also, the auditor has a global view of the organization's data and processes. A crowd auditing system should preserve this complementary division of labor and its affordances.

Informed by the preliminary studies, I developed CrowdTrace, a software prototype to support crowd auditing. CrowdTrace visualizes the structure and provenance of crowd analyses and provides support for 1) identifying problems and 2) creating microtasks. The system's goal is to help auditors in providing actionable feedback and steering crowdsourced refinement of the analyses, thereafter enabling iterative crowdsourced sensemaking.

We evaluated CrowdTrace in a user study under the scenario of refining crowd analyses of a mystery about a fictional terrorist attack. The crowd analysis was generated by CrowdIA in Research Question 3. We measure participants' performance by 1) comparing the problems identified to a gold standard list of important problems, and 2) recruiting crowd workers to evaluate the quality of the microtasks created by each participant. Using CrowdTrace, the

participants were able to focus on the important problems, identifying an average of 13 out of the 26 gold standard problems. Meanwhile, all the important problems were identified by at least one participant (Figure 6.6). Meanwhile, some problems were more successfully identified than others. In terms of creating microtasks, the 19 participants created 110 microtasks in total on CrowdTrace. As a result, we created 110 HITs on MTurk to evaluate each microtask, recruiting a total of 330 crowd workers. Crowd workers found microtasks clear and actionable. When tasks were ambiguous or lacking sufficient context, one third of the crowd workers were able to guess the intention of the task and contribute useful analysis.

Furthermore, we also conducted a content analysis of the participants' crowd auditing processes. We categorized the audit strategies by the direction, how the relevance of information is established, time management between identifying problems and creating microtasks, as well as the styles of annotations. There are pros and cons in the strategies, and the efficacy of each strategy can depend on the auditor's background and experience, but focusing on the known clues and following the leads were the most helpful strategies that led to better auditing results.

To encourage efficiency and learnability, CrowdTrace visualizes the crowd analysis in the pipeline structure and helps non-crowdsourcing-expert auditors to get started easily. I also found that auditors became faster at crowd auditing over time. While auditors recognized that identifying problems and creating microtasks are both difficult, they considered CrowdTrace helpful for both tasks and found it beneficial to have crowd results available for the overall analysis outcome.

Crowd auditing does not only focus on evaluating crowd performance and improving the design of a single microtask, but also emphasize on probing the problems in an existing pipeline of crowd results and steering the refinement with a global context. The crowd analysis is additional information overload in such analysis but also serves as a helpful reference point

for the overall sensemaking process. Clear and detailed descriptions of the problems are important for meaningful and effective hand-off between auditors and crowds but would cost more effort from the auditors, and sometimes is equivalent to fixing the problems directly. These are trade-offs that need to considered when designing crowd auditing systems in a different context and the researchers need to make their careful choices. In addition, the communication between auditors and crowds is also a critical element in crowd auditing. Our results showed that microtask creation can be an effective communication artifact to help auditors steer the crowd refining process.

7.2 Contributions and Broader Implications

My dissertation makes the following contributions:

1. The introduction of *Context Slices* as a collaboration mechanism for bringing non-expert crowds into the sensemaking loop of expert analysts.

2. The *Connect the Dot* software, which uses context slices and crowdsourcing to construct and visualize an entity relationship network from text documents.

3. A *Crowd Sensemaking Pipeline*, which modularizes the sensemaking loop into a series of clearly defined Step Inputs and Step Outputs, and each step implements *Context Slices* to be separately investigated by crowd workers.

4. The *CrowdIA* software, a system that facilitates execution of the *Crowd Sensemaking Pipeline*, with built-in API with Amazon Mechanical Turk to crowdsource sensemaking tasks.

5. A study of the errors and bottlenecks that occur in holistic, unsupervised problem

solving with the *Crowd Sensemaking Pipeline*, analyzing crowd performance in each of the sensemaking steps and the hand-offs between them, and identifying the trade-offs of the amount of local data context in microtasks.

6. A set of design implications for crowdsourced sensemaking systems that chain multiple crowd processes.

7. A novel approach, crowd auditing, to address the challenge of mixed-quality results in a crowd sensemaking pipeline.

8. A crowd auditing system, CrowdTrace, that enables novice auditors to identify problems and create microtasks to refine existing crowd analyses.

9. A set of design implications for crowd auditing tools and iterative crowdsourced sensemaking.

7.3 Future Work

This dissertation presents initial steps towards sensemaking at scale. The research I presented focused on establishing a framework to scaffold crowdsourced sensemaking. The scenario used in the completed work is mystery-solving, which simulates intelligence analysis. While the theoretical basis and data set development have been designed for generalizability, more work is needed to understand how sensemaking processes can be decentralized among different agents in different problems. Specifically, future work is needed to improve the pipeline design and application, as well as adapting the pipeline for different investigation goals and audiences. Another interesting future direction to explore is how artificial intelligence can be introduced in the pipeline to enable mixed-initiative decentralized sensemaking.

7.3.1 Re-examining the Design Decisions

The pipeline modularizes the holistic expert sensemaking process that includes transforming raw data documents to a final presentation of the conclusions and iterative refinement. The design decisions provided a proof of concept for the feasibility of the pipeline, but future work is needed to further optimize and/or adapt the pipeline design.

Context slicing methods. The modularization of the information, namely the *context slices*, provides rich future research opportunities. In Chapter 3, we evaluated 2 context slicing methods: 1) *native slicing* (randomly grouping documents with given a slice size); 2) *co-occurrence slicing* (grouping documents based on keyword co-occurrence). The second method can be formulated as *similarity slicing*, where the similarity metrics can be co-occurrence, frequency-inverse document frequency statistics (tf-idf), or more advanced clustering algorithms. These slicing methods are algorithmic. A bigger research question is *who should slice the context?* Subcontracting crowd work [142] and Alloy [28] have demonstrated the utility of crowd workers or machine learning models to help distribute the work in homogeneous tasks. How can crowds and/or AI be employed in the different steps in the crowd sensemaking pipeline and exploratory analysis? For example, can we train a relevance classifier with a gold-standard dataset and invite crowds to improve the context slices? How would different AI models influence the crowd and the final context slices? How much improvement can crowds make in the context slicing on top of the AI results, if at all? Another question is, how to enable crowd workers to do context slicing? For example, after a crowd worker rated the relevance of one document, we can recommend the next document to rate based on the explanation provided by the worker. My work in Chapter 6 also poses a new question about *how to re-slice the context*. At the early stage of sensemaking, there is no sufficient information to guide context slicing and can result in crowd errors due to lack of

context. Crowd auditing creates new opportunities to re-slice the context with a global view of the data and crowd analyses. Auditors in the studies of Chapter 6 demonstrated a strong advantage in re-distributing the information optimally to elicit the missing pieces. Future research is needed to explore how to utilize the crowd analysis in iterative sensemaking to direct context slicing for more specific goals.

Alternative workflows. The modularization of the process allows for great flexibility of the sequences and workflows to execute the steps. My studies in Chapters 3 and 4 executed the pipeline from the bottom up without any loops between adjacent steps. In expert sensemaking processes, there are usually iterations among the adjacent steps while progressing to later steps, and sometimes certain steps could be merged or skipped. Iterating on local analysis in each step could eliminate local errors and therefore reduce error propagate and enhance the overall analysis outcome. However, this also risks premature focus on less important clues or even irrelevant information to mislead the overall analysis. Future work is needed to further explore how local iterations can be supported in crowdsourced sensemaking. When and how to decide to iterate within local steps? How to keep the analysis up-to-date when there are multiple local iterations in parallel?

Another important problem is how to balance the optimization of local results versus the overall analysis progress. In Chapter 6, I investigated a top-down refining path. Moving forward, an important problem is how the new results can be included in the existing analysis. When the relevant documents (Step 1 Output) are changed, the new results need to go through the remaining steps in the pipeline. How should we distribute the new data as microtasks? We can assign tasks to analyze the new results only or combine the new results with relevant old results. A challenge is that each step would have new results after the crowds addressed the feedback from the auditor. More research is needed to explore how

to manage the new results in different steps and support long-term and efficient iterative sensemaking with crowds. I also hope to explore possibilities of crowds and algorithms collaborating to adjust the pipeline workflows dynamically based on the analysis status.

Optimizing individual steps. The modularized pipeline opens up the sensemaking process as a testbed environment for researchers to design and evaluate novel interfaces for each step. The steps can be replaced or combined with other methods proposed in prior research. Each step can also be individually optimized and plugged back in without influencing other steps. For example, one of the most challenging sensemaking tasks for novice crowds was *Step 2: Read and Extract*. CrowdIA implements a typical create-review workflow in Step 2, but this does not eliminate all poorly extracted information pieces. Reviewer crowds can easily game the task by rating the given information pieces as fulfilling the requirement without a close examination. More complicated task designs such as gated instructions [128] or alternative local workflows can be designed to help crowds better understand the task. For example, instead of asking one crowd worker to create and another worker to review the information pieces, we can ask each crowd worker to first create information pieces for the given documents then peer review, to help them understand the task while eliminating redundancy. Future work is needed to further explore the pros and cons of the alternative local workflows, and the optimization for different applications.

More scalable crowd auditing. The techniques introduced in the crowd sensemaking pipeline open up expert sensemaking processes to large-scale collaboration among novices. The evaluation of errors and bottlenecks furthers our understanding of the opportunities and limitations of incorporating crowdsourcing efforts into complex problem-solving. Crowd auditing makes an important step towards completing the "sensemaking loop" to support correcting analysis mistakes. CrowdTrace assists individual analysts to identify crowd errors

and guide the crowd to refine existing analyses. Given the knowledge of how committed analysts evaluate the bottom-up building path analysis, future research can explore how to facilitate crowd workers in crowd auditing. Prior works have leveraged crowds to break down bigger tasks into smaller ones for other crowd workers [143], and using crowds to provide feedback [67]. The challenge for crowd auditing is to enable crowd workers to oversee the entire pipeline of analysis and provide quality feedback for improving the given analysis. Besides hiring crowd workers to conduct the entire auditing process, we can also explore how to modularize the crowd auditing process to enable micro contributions. The results in Chapter 6 can serve as a baseline to evaluate crowd performance in crowd auditing.

Broader applications. Meanwhile, solving mysteries is just one application of decentralized intelligence. Another promising direction is to apply the crowd sensemaking pipeline to dynamic datasets such as social media and online communities, and a broader set of analysis goals such as fact-checking and understanding public opinions. I am also interested in studying if and how the knowledge and context transfer when the crowd overlaps in multiple analysis processes. I plan to release CrowdIA and CrowdTrace for public use and conduct field studies to investigate more applications of the pipeline in real-world problems. It will also be interesting to study how different crowds and different levels of involvement would influence the sensemaking outcome.

7.3.2 Mixed-Initiative Intelligent Systems

Moving forward, I aim to build technologies to empower human intelligence in greater depth. Now that CrowdIA successfully decomposes the big black box of expert sensemaking processes into multiple smaller black boxes of crowd sensemaking tasks, I seek to further leverage AI techniques in the sensemaking pipeline.

The pipeline can serve as a framework to connect multiple algorithms and/or models. In the machine learning field, researchers have made key progress in reading comprehension QA tasks, building the model to "reason, gather, and synthesize disjoint pieces of information within the context to generate an answer" (*NarrativeQA*) [172]. The challenge for using those methods for mystery solving, however, is the lack of training data. One of the exciting directions I hope to explore is to compare the pipeline of analysis by crowds and off-the-shelf models and develop human-AI teaming solutions for sensemaking.

Building on this idea, another exciting direction is to compensate for the lack of training data by using crowds to teach sensemaking processes to machines. Crowdsourced *machine teaching* has the potential to leverage a broad range of human knowledge and experiences to develop creative solutions when training data is not available. Crowdsourcing is also advantageous with the potential to combine teaching and training to build smarter AI systems.

I am also interested in the broader domain of human-AI interaction and teaming. In Chapter 5, we modeled the errors and bottlenecks in crowd sensemaking. Specifically, we contributed a typology of local errors in exploratory analyses. There is also prior research that attempted to learn worker behavioral traces [159] to predict crowd performance. Building on these results, future work can explore how to use this insight to automatically provide feedback to crowd workers based on their answers in real-time. For example, "goode" or other single word responses are typical low-effort indicators. If the crowd workers input such answers in the microtask, the task UI can show a warning of potential rejection due to low-quality work. This is analogous to a knowledge-based recommendation system. In the long run, smarter models can be trained based on requester evaluation and classification of the crowd results and errors, to assess the quality of crowd work and make suggestions to re-examine the results before the crowd workers submit their results. Certainly, this brings up a new design problem for crowd-machine teaming: how to help the crowd in a positive way

without introducing extra workload or unpleasant working experience, and how to scaffold the communication between crowd workers and the AI models?

CrowdIA breaks down the complicated human sensemaking process into connected, modularized steps. An intriguing future research question is, can we use the pipeline as a framework to break down and interpret the existing sensemaking results, such as online discourse on social media, or predictions made by neural networks? Another problem I am interested in pursuing in my future research is designing customized explainable interfaces for different audiences. Data security analysts across different organizations have diverse security policies and concerns. Machine learning engineers, analysts, and domain experts are interested in different stages of model training and have different goals. Different audiences need different perspectives and granularities in the explanations of the same AI decisions. How can we customize explainable AI for diverse users? Furthermore, human behaviors are influenced by AI decisions, especially with the increasing public awareness of, experiences with, and diverse attitudes towards AI technologies. My ongoing work with Microsoft Research (MSR) aims to evaluate the user experience (UX) impact of the guidelines for human-AI interaction [3]. Another ongoing project in collaboration with MSR and Oregon State University aims to understand the impact of AI on problem-solving experiences for users of different genders. I will continue these research efforts to contribute to the development of more ethical and responsible AI technologies.

Bibliography

[1] Mike's place, May 2020. URL https://en.wikipedia.org/wiki/Mike's_Place.

[2] Omar Alonso, Daniel E. Rose, and Benjamin Stewart. Crowdsourcing for relevance evaluation. *SIGIR Forum*, 42(2):9–15, November 2008. ISSN 0163-5840. doi: 10.1145/1480506.1480508. URL https://doi.org/10.1145/1480506.1480508.

[3] Saleema Amershi, Dan Weld, Mihaela Vorvoreanu, Adam Fourney, Besmira Nushi, Penny Collisson, Jina Suh, Shamsi Iqbal, Paul N Bennett, Kori Inkpen, et al. Guidelines for human-ai interaction. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pages 1–13, 2019.

[4] Salvatore Andolina, Hendrik Schneider, Joel Chan, Khalil Klouche, Giulio Jacucci, and Steven Dow. Crowdboard: Augmenting in-person idea generation with real-time crowds. In *Proceedings of the 2017 ACM SIGCHI Conference on Creativity and Cognition*, C&C '17, pages 106–118, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-4403-6. doi: 10.1145/3059454.3059477. URL http://doi.acm.org/10.1145/3059454.3059477.

[5] Paul André, Aniket Kittur, and Steven P. Dow. Crowd synthesis: Extracting categories and clusters from complex data. In *Proceedings of the 17th ACM Conference on Computer Supported Cooperative Work Social Computing*, CSCW '14, page 989–998, New York, NY, USA, 2014. Association for Computing Machinery. ISBN 9781450325400. doi: 10.1145/2531602.2531653. URL https://doi.org/10.1145/2531602.2531653.

[6] Paul André, Aniket Kittur, and Steven P. Dow. Crowd synthesis: Extracting categories and clusters from complex data. In *Proceedings of the 17th ACM Conference on*

Computer Supported Cooperative Work & Social Computing, CSCW '14, pages 989–998, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2540-0. doi: 10.1145/2531602.2531653. URL http://doi.acm.org/10.1145/2531602.2531653.

[7] Ahmer Arif, John J Robinson, Stephanie A Stanek, Elodie S Fichet, Paul Townsend, Zena Worku, and Kate Starbird. A closer look at the self-correcting crowd: Examining corrections in online rumors. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing*, pages 155–168. ACM, 2017.

[8] Tal August and Katharina Reinecke. Pay attention, please: Formal language improves attention in volunteer and paid online experiments. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450359702. doi: 10.1145/3290605.3300478. URL https://doi.org/10.1145/3290605.3300478.

[9] Tal August and Katharina Reinecke. Pay attention, please: Formal language improves attention in volunteer and paid online experiments. 2019.

[10] L. Bavoil, S. P. Callahan, P. J. Crossno, J. Freire, C. E. Scheidegger, C. T. Silva, and H. T. Vo. Vistrails: enabling interactive multiple-view visualizations. In *VIS 05. IEEE Visualization, 2005.*, pages 135–142, Oct 2005. doi: 10.1109/VISUAL.2005.1532788.

[11] Michael S Bernstein, Greg Little, Robert C Miller, Björn Hartmann, Mark S Ackerman, David R Karger, David Crowell, and Katrina Panovich. Soylent: a word processor with a crowd inside. In *Proceedings of the 23nd annual ACM symposium on User interface software and technology*, pages 313–322, New York, New York, USA, 2010. ACM Press. ISBN 9781450302715. doi: 10.1145/1866029.1866078. URL http://portal.acm.org/citation.cfm?doid=1866029.1866078.

[12] Michael S. Bernstein, Joel Brandt, Robert C. Miller, and David R. Karger. Crowds in Two Seconds: Enabling Realtime Crowd-powered Interfaces. UIST '11, pages 33–42, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0716-1. doi: 10.1145/2047196.2047201. URL http://doi.acm.org/10.1145/2047196.2047201http://dl.acm.org/ft_gateway.cfm?id=2047201&type=pdf.

[13] Michael S. Bernstein, Joel Brandt, Robert C. Miller, and David R. Karger. Crowds in two seconds: Enabling realtime crowd-powered interfaces. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, UIST '11, pages 33–42, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0716-1. doi: 10.1145/2047196.2047201. URL http://doi.acm.org/10.1145/2047196.2047201.

[14] Eric A. Bier, Stuart K. Card, and John W. Bodnar. Entity-based collaboration tools for intelligence analysis. In *2008 IEEE Symposium on Visual Analytics Science and Technology*, pages 99–106. IEEE, 10 2008. ISBN 978-1-4244-2935-6. doi: 10.1109/VAST.2008.4677362. URL http://ieeexplore.ieee.org/document/4677362/http://ieeexplore.ieee.org/ielx5/4669530/4677336/04677362.pdf?tp=&arnumber=4677362&isnumber=4677336.

[15] Eric A Bier, Stuart K Card, and John W Bodnar. Entity-based collaboration tools for intelligence analysis. In *2008 IEEE Symposium on Visual Analytics Science and Technology*, pages 99–106. IEEE, 2008.

[16] Jeffrey P. Bigham, Chandrika Jayant, Hanjie Ji, Greg Little, Andrew Miller, Robert C. Miller, Robin Miller, Aubrey Tatarowicz, Brandyn White, Samual White, and Tom Yeh. Vizwiz: Nearly real-time answers to visual questions. In *Proceedings of the 23Nd Annual ACM Symposium on User Interface Software and Technology*, UIST

'10, pages 333–342, New York, NY, USA, 2010. ACM. ISBN 978-1-4503-0271-5. doi: 10.1145/1866029.1866080. URL http://doi.acm.org/10.1145/1866029.1866080.

[17] Alessandro Bozzon, Marco Brambilla, and Stefano Ceri. Answering search queries with crowdsearcher. In *Proceedings of the 21st International Conference on World Wide Web*, WWW '12, pages 1009–1018, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1229-5. doi: 10.1145/2187836.2187971. URL http://doi.acm.org/10.1145/2187836.2187971.

[18] Lauren Bradel, Alex Endert, Kristen Koch, Christopher Andrews, and Chris North. Large high resolution displays for co-located collaborative sensemaking: Display usage and territoriality. *International Journal of Human-Computer Studies*, 71(11):1078–1088, 2013.

[19] Jonathan Bragg, Mausam, and Daniel S. Weld. Optimal testing for crowd workers. In *Proceedings of the 2016 International Conference on Autonomous Agents Multiagent Systems*, AAMAS '16, page 966–974, Richland, SC, 2016. International Foundation for Autonomous Agents and Multiagent Systems. ISBN 9781450342391.

[20] Jonathan Bragg, Mausam, and Daniel S. Weld. Sprout: Crowd-powered task design for crowdsourcing. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*, UIST '18, page 165–176, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450359481. doi: 10.1145/3242587.3242598. URL https://doi.org/10.1145/3242587.3242598.

[21] Chris Callison-Burch. Fast, cheap, and creative: Evaluating translation quality using amazon's mechanical turk. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1 - Volume 1*, EMNLP '09, page 286–295, USA, 2009. Association for Computational Linguistics. ISBN 9781932432596.

[22] Chris Callison-Burch. Fast, cheap, and creative: evaluating translation quality using amazon's mechanical turk. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1*, pages 286–295. Association for Computational Linguistics, 2009.

[23] L. Elisa Celis, Sai Praneeth Reddy, Ishaan Preet Singh, and Shailesh Vaya. Assignment Techniques for Crowdsourcing Sensitive Tasks. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing*, CSCW '16, pages 836–847, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-3592-8. doi: 10.1145/2818048.2835202. URL http://doi.acm.org/10.1145/2818048.2835202.

[24] Joel Chan, Steven Dang, and Steven P. Dow. IdeaGens: Enabling Expert Facilitation of Crowd Brainstorming. pages 13–16, 2016. ISBN 9781450339506. doi: 10.1145/2818052.2874313. URL http://dl.acm.org/citation.cfm?doid=2818052.2874313.

[25] Joel Chan, Steven Dang, and Steven P. Dow. Improving crowd innovation with expert facilitation. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing*, CSCW '16, pages 1223–1235, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-3592-8. doi: 10.1145/2818048.2820023. URL http://doi.acm.org/10.1145/2818048.2820023.

[26] Joel Chan, Steven Dang, and Steven P. Dow. Comparing Different Sensemaking Approaches for Large-Scale Ideation. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, pages 2717–2728, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-3362-7. doi: 10.1145/2858036.2858178. URL http://doi.acm.org/10.1145/2858036.2858178.

[27] Joseph Chee Chang, Aniket Kittur, and Nathan Hahn. Alloy: Clustering with crowds and computation. In *Proceedings of the 2016 CHI Conference on Human Factors in*

Computing Systems, CHI '16, page 3180–3191, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450333627. doi: 10.1145/2858036.2858411. URL https://doi.org/10.1145/2858036.2858411.

[28] Joseph Chee Chang, Aniket Kittur, and Nathan Hahn. Alloy: Clustering with crowds and computation. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems - CHI '16*, pages 3180–3191, New York, New York, USA, 2016. ACM Press. ISBN 9781450333627. doi: 10.1145/2858036.2858411. URL http://dl.acm.org/citation.cfm?doid=2858036.2858411.

[29] Joseph Chee Chang, Aniket Kittur, and Nathan Hahn. Alloy: Clustering with crowds and computation. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, pages 3180–3191, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-3362-7. doi: 10.1145/2858036.2858411. URL http://doi.acm.org/10.1145/2858036.2858411.

[30] Wen-Huang Cheng and David Gotz. Context-based page unit recommendation for web-based sensemaking tasks. In *Proceedingsc of the 13th international conference on Intelligent user interfaces - IUI '09*, page 107, New York, New York, USA, 2008. ACM Press. ISBN 9781605581682. doi: 10.1145/1502650.1502668. URL http://portal.acm.org/citation.cfm?doid=1502650.1502668.

[31] Wen-Huang Cheng and David Gotz. Context-based page unit recommendation for web-based sensemaking tasks. In *Proceedings of the 14th International Conference on Intelligent User Interfaces*, IUI '09, page 107–116, New York, NY, USA, 2009. Association for Computing Machinery. ISBN 9781605581682. doi: 10.1145/1502650.1502668. URL https://doi.org/10.1145/1502650.1502668.

[32] Wen-Huang Cheng and David Gotz. Context-based page unit recommendation for

web-based sensemaking tasks. In *Proceedings of the 14th International Conference on Intelligent User Interfaces*, IUI '09, pages 107–116, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-168-2. doi: 10.1145/1502650.1502668. URL http://doi.acm.org/10.1145/1502650.1502668.

[33] Lydia B. Chilton, Greg Little, Darren Edge, Daniel S. Weld, and James A. Landay. Cascade: Crowdsourcing taxonomy creation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, page 1999–2008, New York, NY, USA, 2013. Association for Computing Machinery. ISBN 9781450318990. doi: 10.1145/2470654.2466265. URL https://doi.org/10.1145/2470654.2466265.

[34] Lydia B. Chilton, Greg Little, Darren Edge, Daniel S. Weld, and James A. Landay. Cascade: Crowdsourcing Taxonomy Creation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI '13*, page 1999, New York, New York, USA, 2013. ACM Press. ISBN 9781450318990. doi: 10.1145/2470654.2466265. URL http://dl.acm.org/citation.cfm?doid=2470654.2466265.

[35] Lydia B. Chilton, Greg Little, Darren Edge, Daniel S. Weld, and James A. Landay. Cascade: Crowdsourcing taxonomy creation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, pages 1999–2008, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-1899-0. doi: 10.1145/2470654.2466265. URL http://doi.acm.org/10.1145/2470654.2466265.

[36] Lydia B Chilton, Juho Kim, Paul André, Felicia Cordeiro, James A Landay, Daniel S Weld, Steven P Dow, Robert C Miller, and Haoqi Zhang. Frenzy: collaborative data organization for creating conference sessions. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems*, pages 1255–1264. ACM, 2014.

[37] George Chin, Olga A. Kuchar, and Katherine E. Wolf. Exploring the analyti-

cal processes of intelligence analysts. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '09, page 11–20, New York, NY, USA, 2009. Association for Computing Machinery. ISBN 9781605582467. doi: 10.1145/1518701.1518704. URL https://doi.org/10.1145/1518701.1518704.

[38] George Chin, Jr., Olga A. Kuchar, and Katherine E. Wolf. Exploring the analytical processes of intelligence analysts. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '09, pages 11–20, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-246-7. doi: 10.1145/1518701.1518704. URL http://doi.acm.org/10.1145/1518701.1518704.

[39] George Jr. Chin, Olga A. Kuchar, and Katherine E. Wolf. Exploring the Analytical Processes of Intelligence Analysts. In *Chi '09*, CHI '09, pages 11–20, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-246-7. doi: 10.1145/1518701.1518704. URL http://doi.acm.org/10.1145/1518701.1518704http://dl.acm.org/ft_gateway.cfm?id=1518704&type=pdfhttp://dl.acm.org/citation.cfm?doid=1518701.1518704http://portal.acm.org/citation.cfm?doid=1518701.1518704.

[40] H. Chung, S. P. Dasari, S. Nandhakumar, and C. Andrews. Cricto: Supporting sense-making through crowdsourced information schematization. In *2017 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 139–150, Oct 2017. doi: 10.1109/VAST.2017.8585484.

[41] Haeyong Chung, Seungwon Yang, Naveed Massjouni, Christopher Andrews, Rahul Kanna, and Chris North. VizCept: Supporting synchronous collaboration for constructing visualizations in intelligence analysis. pages 107–114, 10 2010. ISBN 9781424494866. doi: 10.1109/VAST.2010.5652932. URL http://ieeexplore.ieee.

org/document/5652932/http://ieeexplore.ieee.org/ielx5/5638583/5649053/05652932.pdf?tp=&arnumber=5652932&isnumber=5649053.

[42] Haeyong Chung, Seungwon Yang, Naveed Massjouni, Christopher Andrews, Rahul Kanna, and Chris North. Vizcept: Supporting synchronous collaboration for constructing visualizations in intelligence analysis. In *2010 IEEE Symposium on Visual Analytics Science and Technology*, pages 107–114. IEEE, 2010.

[43] Robert M Clark. *Intelligence analysis: a target-centric approach*. CQ Press, Thousand Oaks, Calif, 4th edition, 2013. ISBN 1452206120;9781452206127;.

[44] Kristin A Cook and James J Thomas. Illuminating the path: The research and development agenda for visual analytics. Technical report, Pacific Northwest National Lab.(PNNL), Richland, WA (United States), 2005.

[45] Justin Cranshaw and Aniket Kittur. The polymath project: Lessons from a successful online collaboration in mathematics. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 1865–1874, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0228-9. doi: 10.1145/1978942.1979213. URL http://doi.acm.org/10.1145/1978942.1979213.

[46] Arthur Cropley and David Cropley. Resolving the paradoxes of creativity: an extended phase model. *Cambridge Journal of Education*, 38(3):355–373, 2008. doi: 10.1080/03057640802286871. URL https://doi.org/10.1080/03057640802286871.

[47] R. Jordon Crouser and Remco Chang. An Affordance-Based Framework for Human Computation and Human-Computer Collaboration. volume 18, pages 2859–2868, 12 2012. doi: 10.1109/TVCG.2012.195. URL http://ieeexplore.ieee.org/abstract/document/6327292/http://ieeexplore.

ieee.org/ielx5/2945/6327196/06327292.pdf?tp=&arnumber=6327292&isnumber=6327196http://dx.doi.org/10.1109/TVCG.2012.195.

[48] Peng Dai, Mausam, and Daniel S. Weld. Decision-theoretic control of crowd-sourced workflows. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence*, AAAI'10, pages 1168–1174. AAAI Press, 2010. URL http://dl.acm.org/citation.cfm?id=2898607.2898793.

[49] Ofer Dekel and Ohad Shamir. Vox populi: Collecting high-quality labels from a crowd.

[50] Gianluca Demartini, Djellel Eddine Difallah, and Philippe Cudré-Mauroux. Zencrowd: Leveraging probabilistic reasoning and crowdsourcing techniques for large-scale entity linking. In *Proceedings of the 21st International Conference on World Wide Web*, WWW '12, pages 469–478, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1229-5. doi: 10.1145/2187836.2187900. URL http://doi.acm.org/10.1145/2187836.2187900.

[51] doggyxp. Braingle: 'the case of the mischief maker' brain teaser, 2018. URL http://www.braingle.com/brainteasers/17325/the-case-of-the-mischief-maker.html.

[52] Steven Dow, Anand Kulkarni, Scott Klemmer, and Björn Hartmann. Shepherding the crowd yields better work. In *Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work*, CSCW '12, page 1013–1022, New York, NY, USA, 2012. Association for Computing Machinery. ISBN 9781450310864. doi: 10.1145/2145204.2145355. URL https://doi.org/10.1145/2145204.2145355.

[53] Steven Dow, Anand Kulkarni, Scott Klemmer, and Björn Hartmann. Shepherding the crowd yields better work. In *Proceedings of the ACM 2012 Conference on Computer

Supported Cooperative Work, CSCW '12, pages 1013–1022, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1086-4. doi: 10.1145/2145204.2145355. URL http://doi.acm.org/10.1145/2145204.2145355.

[54] Steven Dow, Anand Kulkarni, Scott Klemmer, and Björn Hartmann. Shepherding the crowd yields better work. In *Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work*, CSCW '12, pages 1013–1022, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1086-4. doi: 10.1145/2145204.2145355. URL http://doi.acm.org/10.1145/2145204.2145355.

[55] Julie S Downs, Mandy B Holbrook, Steve Sheng, and Lorrie Faith Cranor. Are your participants gaming the system?: screening mechanical turk workers. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 2399–2402. ACM, 2010.

[56] Philipp Drieger. Semantic Network Analysis as a Method for Visual Text Analytics. volume 79, pages 4–17. Elsevier B.V., 2013. ISBN 1877-0428. doi: 10.1016/j.sbspro.2013.05.053. URL http://linkinghub.elsevier.com/retrieve/pii/S1877042813010227.

[57] Carsten Eickhoff and Arjen de Vries. How crowdsourcable is your task. In *Proceedings of the workshop on crowdsourcing for search and data mining (CSDM) at the fourth ACM international conference on web search and data mining (WSDM)*, pages 11–14, 2011.

[58] Alex Endert, Patrick Fiaux, and Chris North. Semantic interaction for visual text analytics. pages 473–482, 2012. ISBN 9781450310154. doi: 10.1145/2207676.2207741. URL http://delivery.acm.org/10.1145/2210000/2207741/p473-endert.pdf?ip=128.173.38.42&id=2207741&acc=ACTIVESERVICE&key=B33240AC40EC9E30.

80AE0C8B3B97B250.4D4702B0C3E38B35.4D4702B0C3E38B35&CFID=828948590&CFTOKEN=61461544&__acm__=1510587030_9e7f21ea71bc63ac4f3a9.

[59] Tim Finin, Will Murnane, Anand Karandikar, Nicholas Keller, Justin Martineau, and Mark Dredze. Annotating named entities in twitter data with crowdsourcing. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, CSLDAMT '10, pages 80–88, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics. URL http://dl.acm.org/citation.cfm?id=1866696.1866709.

[60] Jenny Rose Finkel, Trond Grenager, and Christopher Manning. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd annual meeting on association for computational linguistics*, pages 363–370. Association for Computational Linguistics, 2005.

[61] Kristie Fisher, Scott Counts, and Aniket Kittur. Distributed sensemaking: Improving sensemaking by leveraging the efforts of previous users. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, page 247–256, New York, NY, USA, 2012. Association for Computing Machinery. ISBN 9781450310154. doi: 10.1145/2207676.2207711. URL https://doi.org/10.1145/2207676.2207711.

[62] Kristie Fisher, Scott Counts, and Aniket Kittur. Distributed sensemaking: Improving sensemaking by leveraging the efforts of previous users. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 247–256, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1015-4. doi: 10.1145/2207676.2207711. URL http://doi.acm.org/10.1145/2207676.2207711.

[63] Radu Florea and Ramona Florea. Audit techniques and audit evidence. *Economy Transdisciplinarity Cognition*, 14(1), 2011.

[64] Brooke Foucault Welles and Weiai Xu. Network visualization and problem-solving support: A cognitive fit study. volume 54, pages 162–167. North-Holland, 7 2018. doi: 10.1016/J.SOCNET.2018.01.005. URL https://www.sciencedirect.com/science/inproceedings/pii/S0378873317301740?via%3Dihub.

[65] Ujwal Gadiraju, Jie Yang, and Alessandro Bozzon. Clarity is a worthwhile quality: On the role of task clarity in microtask crowdsourcing. In *Proceedings of the 28th ACM Conference on Hypertext and Social Media*, HT '17, page 5–14, New York, NY, USA, 2017. Association for Computing Machinery. ISBN 9781450347082. doi: 10.1145/3078714.3078715. URL https://doi.org/10.1145/3078714.3078715.

[66] Yang Gao, Yan Chen, and KJ Ray Liu. On cost-effective incentive mechanisms in microtask crowdsourcing. *IEEE Trans. Comput. Intellig. and AI in Games*, 7(1):3–15, 2015.

[67] D Gary. Patty wagstaff's second act. volume 26, pages 20–25. Smithsonian Business Ventures, 2011.

[68] Daniel Gigone and Reid Hastie. The common knowledge effect: Information sharing and group judgment. *Journal of Personality and social Psychology*, 65(5):959, 1993.

[69] Steven Glover. Analytical procedures and audit planning decisions: Unexpected fluctuations that influence auditors to revise their audit plans. *Journal of Accountancy*, 191:99, 02 2001.

[70] Steven Gottlieb, Sheldon I Arenberg, Raj Singh, et al. *Crime analysis: From first report to final arrest*. Alpha Publishing Montclair, CA, 1994.

[71] D. Gotz, M. X. Zhou, and V. Aggarwal. Interactive visual synthesis of analytic knowl-

edge. In *2006 IEEE Symposium On Visual Analytics Science And Technology*, pages 51–58, Oct 2006. doi: 10.1109/VAST.2006.261430.

[72] Nitesh Goyal and Susan R. Fussell. Effects of Sensemaking Translucence on Distributed Collaborative Analysis. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing - CSCW '16*, pages 287–301, New York, New York, USA, 2016. ACM Press. ISBN 9781450335928. doi: 10.1145/2818048.2820071. URL http://dl.acm.org/citation.cfm?doid=2818048.2820071http://dl.acm.org/citation.cfm?id=2818048.2820071.

[73] Nitesh Goyal and Susan R Fussell. Effects of sensemaking translucence on distributed collaborative analysis. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing*, pages 288–302. ACM, 2016.

[74] Nitesh Goyal, Gilly Leshed, and Susan R. Fussell. Effects of visualization and note-taking on sensemaking and analysis. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, page 2721–2724, New York, NY, USA, 2013. Association for Computing Machinery. ISBN 9781450318990. doi: 10.1145/2470654.2481376. URL https://doi.org/10.1145/2470654.2481376.

[75] Nitesh Goyal, Gilly Leshed, and Susan R. Fussell. Effects of visualization and note-taking on sensemaking and analysis. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, pages 2721–2724, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-1899-0. doi: 10.1145/2470654.2481376. URL http://doi.acm.org/10.1145/2470654.2481376.

[76] Catherine Grady and Matthew Lease. Crowdsourcing document relevance assessment with mechanical turk. In *Proceedings of the NAACL HLT 2010 workshop on creating*

speech and language data with Amazon's mechanical turk, pages 172–179. Association for Computational Linguistics, 2010.

[77] Mary L Gray and Siddharth Suri. *Ghost Work: How to Stop Silicon Valley from Building a New Global Underclass*. Eamon Dolan Books, 2019.

[78] Catherine Grevet and Eric Gilbert. Piggyback prototyping: Using existing, large-scale social computing systems to prototype new ones. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, pages 4047–4056, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3145-6. doi: 10.1145/2702123.2702395. URL http://doi.acm.org/10.1145/2702123.2702395.

[79] Philipp Gutheim and Björn Hartmann. Fantasktic: Improving quality of results for novice crowdsourcing users. *EECS Dept., Univ. California, Berkeley, CA, USA, Tech. Rep. UCB/EECS-2012-112*, 2012.

[80] Nathan Hahn, Joseph Chang, Ji Eun Kim, and Aniket Kittur. The knowledge accelerator: Big picture thinking in small pieces. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, page 2258–2270, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450333627. doi: 10.1145/2858036.2858364. URL https://doi.org/10.1145/2858036.2858364.

[81] Nathan Hahn, Joseph Chang, Ji Eun Kim, and Aniket Kittur. The Knowledge Accelerator: Big Picture Thinking in Small Pieces. pages 2258–2270, 2016. ISBN 9781450333627. doi: 10.1145/2858036.2858364. URL http://dl.acm.org/citation.cfm?doid=2858036.2858364.

[82] Nathan Hahn, Joseph Chang, Ji Eun Kim, and Aniket Kittur. The knowledge accelerator: Big picture thinking in small pieces. In *Proceedings of the 2016 CHI Conference*

on Human Factors in Computing Systems, CHI '16, pages 2258–2270, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-3362-7. doi: 10.1145/2858036.2858364. URL http://doi.acm.org/10.1145/2858036.2858364.

[83] Michael G Harvey and Robert F Lusch. Expanding the nature and scope of due diligence. *Journal of Business Venturing*, 10(1):5–21, 1995.

[84] Jeffrey Heer, Fernanda B. Viégas, and Martin Wattenberg. Voyagers and voyeurs. In *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '07*, page 1029, New York, New York, USA, 2007. ACM Press. ISBN 9781595935939. doi: 10.1145/1240624.1240781. URL http://portal.acm.org/citation.cfm?doid=1240624.1240781.

[85] David Dryden Henningsen, Mary Lynn Miller Henningsen, Jennifer Eden, and Michael G Cruz. Examining the symptoms of groupthink and retrospective sense-making. *Small Group Research*, 37(1):36–64, 2006.

[86] Richards J Heuer. *Psychology of intelligence analysis*. Jeffrey Frank Jones, 1999.

[87] Richards J. Heuer and Center for the Study of Intelligence (U.S.). *Psychology of intelligence analysis*. Lulu. com, 1999. ISBN 1929667000.

[88] Shih-Wen Huang and Wai-Tat Fu. Don't hide in the crowd!: Increasing social transparency between peer workers improves crowdsourcing outcomes. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, pages 621–630, New York, NY, USA, 2013. ACM. ISBN 978-1-4503-1899-0. doi: 10.1145/2470654.2470743. URL http://doi.acm.org/10.1145/2470654.2470743.

[89] F Hughes and D Schum. Discovery-proof-choice, the art and science of the process of intelligence analysis-preparing for the future of intelligence analysis. 2003.

[90] Rogstadius Jakob, Kostakos Vassilis, Kittur Aniket, Smus Boris, Laredo Jim, and Vukovic Maja. An assessment of intrinsic and extrinsic motivation on task performance in crowdsourcing markets, 2011.

[91] Irving Lester Janis and Irving Lester Janis. *Groupthink: Psychological studies of policy decisions and fiascoes*, volume 349. Houghton Mifflin Boston, 1982.

[92] Ruogu Kang, Aimee Kane, and Sara Kiesler. Teammate inaccuracy blindness: When information sharing tools hinder collaborative analysis. In *Proceedings of the 17th ACM Conference on Computer Supported Cooperative Work Social Computing*, CSCW '14, page 797–806, New York, NY, USA, 2014. Association for Computing Machinery. ISBN 9781450325400. doi: 10.1145/2531602.2531681. URL https://doi.org/10.1145/2531602.2531681.

[93] Ruogu Kang, Aimee Kane, and Sara Kiesler. Teammate inaccuracy blindness: When information sharing tools hinder collaborative analysis. In *Proceedings of the 17th ACM Conference on Computer Supported Cooperative Work & Social Computing*, CSCW '14, pages 797–806, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2540-0. doi: 10.1145/2531602.2531681. URL http://doi.acm.org/10.1145/2531602.2531681.

[94] Youn-ah Kang and John Stasko. Characterizing the intelligence analysis process through a longitudinal field study: Implications for visual analytics. *Information Visualization*, 13(2):134–158, 2014.

[95] Joy Kim, Justin Cheng, and Michael S Bernstein. Ensemble: exploring complementary strengths of leaders and crowds in creative collaboration. In *Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing*, pages 745–755. ACM, 2014.

[96] Joy Kim, Sarah Sterman, Allegra Argent Beal Cohen, and Michael S Bernstein. Mechanical Novel: Crowdsourcing Complex Work through Revision. In *Proceedings of the 20th ACM Conference on Computer Supported Cooperative Work & Social Computing*, pages 233–245, 2016. ISBN 9781450343350. doi: 10.1145/2998181.2998196.

[97] Joy Kim, Sarah Sterman, Allegra Argent Beal Cohen, and Michael S. Bernstein. Mechanical novel: Crowdsourcing complex work through reflection and revision. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing*, CSCW '17, page 233–245, New York, NY, USA, 2017. Association for Computing Machinery. ISBN 9781450343350. doi: 10.1145/2998181.2998196. URL https://doi.org/10.1145/2998181.2998196.

[98] Joy Kim, Sarah Sterman, Allegra Argent Beal Cohen, and Michael S Bernstein. Mechanical novel: Crowdsourcing complex work through reflection and revision. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing*, pages 233–245. ACM, 2017.

[99] Peter Kinnaird, Laura Dabbish, and Sara Kiesler. Workflow transparency in a microtask marketplace. In *Proceedings of the 17th ACM international conference on Supporting group work*, pages 281–284. ACM, 2012.

[100] Aniket Kittur, Ed H Chi, and Bongwon Suh. Crowdsourcing user studies with mechanical turk. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 453–456. ACM, 2008.

[101] Aniket Kittur, Boris Smus, Susheel Khamkar, and Robert E. Kraut. Crowdforge: Crowdsourcing complex work. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, UIST '11, page 43–52, New York, NY,

USA, 2011. Association for Computing Machinery. ISBN 9781450307161. doi: 10.1145/2047196.2047202. URL https://doi.org/10.1145/2047196.2047202.

[102] Aniket Kittur, Boris Smus, and Robert Kraut. CrowdForge Crowdsourcing Complex Work. page 1801, 2011. ISBN 9781450302685. doi: 10.1145/1979742.1979902. URL http://www.scopus.com/inward/record.url?eid=2-s2.0-79957956324&partnerID=tZOtx3y1.

[103] Aniket Kittur, Susheel Khamkar, Paul André, and Robert Kraut. Crowdweaver: Visually managing complex crowd work. In *Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work*, CSCW '12, pages 1033–1036, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1086-4. doi: 10.1145/2145204.2145357. URL http://doi.acm.org/10.1145/2145204.2145357.

[104] Aniket Kittur, Andrew M. Peters, Abdigani Diriye, Trupti Telang, and Michael R. Bove. Costs and benefits of structured information foraging. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI '13*, page 2989, New York, New York, USA, 2013. ACM Press. ISBN 9781450318990. doi: 10.1145/2470654.2481415. URL http://dl.acm.org/citation.cfm?doid=2470654.2481415.

[105] Aniket Kittur, Andrew M. Peters, Abdigani Diriye, and Michael Bove. Standing on the schemas of giants: Socially augmented information foraging. In *Proceedings of the 17th ACM Conference on Computer Supported Cooperative Work Social Computing*, CSCW '14, page 999–1010, New York, NY, USA, 2014. Association for Computing Machinery. ISBN 9781450325400. doi: 10.1145/2531602.2531644. URL https://doi.org/10.1145/2531602.2531644.

[106] Aniket Kittur, Andrew M. Peters, Abdigani Diriye, and Michael Bove. Standing on the Schemas of Giants: Socially Augmented Information Foraging. In *Pro-*

ceedings of the 17th ACM conference on Computer supported cooperative work & social computing - CSCW '14, CSCW '14, pages 999–1010, New York, New York, USA, 2014. ACM Press. ISBN 9781450325400. doi: 10.1145/2531602.2531644. URL http://dl.acm.org/citation.cfm?doid=2531602.2531644http://dl.acm.org/citation.cfm?id=2531602.2531644http://doi.acm.org/10.1145/2531602.2531644http://dl.acm.org/ft_gateway.cfm?id=2531644&type=pdf.

[107] Aniket Kittur, Lixiu Yu, Tom Hope, Joel Chan, Hila Lifshitz-Assaf, Karni Gilon, Felicia Ng, Robert E Kraut, and Dafna Shahaf. Scaling up analogical innovation with crowds and ai. *Proceedings of the National Academy of Sciences*, 116(6):1870–1877, 2019.

[108] Gary Klein, Brian Moon, and Robert R Hoffman. Making sense of sensemaking 2: A macrocognitive model. *IEEE Intelligent systems*, 21(5):88–92, 2006.

[109] Gary Klein, Jennifer K Phillips, Erica L Rall, and Deborah A Peluso. A data-frame theory of sensemaking. In *Expertise out of context: Proceedings of the sixth international conference on naturalistic decision making*, pages 113–155. New York, NY, USA: Lawrence Erlbaum, 2007.

[110] Masaki Kobayashi, Hiromi Morita, Masaki Matsubara, Nobuyuki Shimizu, and Atsuyuki Morishima. An empirical study on short-and long-term effects of self-correction in crowdsourced microtasks. In *Sixth AAAI Conference on Human Computation and Crowdsourcing*, 2018.

[111] Todd Kulesza, Saleema Amershi, Rich Caruana, Danyel Fisher, and Denis Charles. Structured labeling for facilitating concept evolution in machine learning. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, page 3075–3084, New York, NY, USA, 2014. Association for Computing Machinery.

ISBN 9781450324731. doi: 10.1145/2556288.2557238. URL https://doi.org/10.1145/2556288.2557238.

[112] Anand Kulkarni, Matthew Can, and Björn Hartmann. Collaboratively crowdsourcing workflows with turkomatic. In *Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work*, CSCW '12, page 1003–1012, New York, NY, USA, 2012. Association for Computing Machinery. ISBN 9781450310864. doi: 10.1145/2145204.2145354. URL https://doi.org/10.1145/2145204.2145354.

[113] Anand Kulkarni, Matthew Can, and Björn Hartmann. Collaboratively crowdsourcing workflows with turkomatic. In *Proceedings of the acm 2012 conference on computer supported cooperative work*, pages 1003–1012. ACM, 2012.

[114] Anand Kulkarni, Matthew Can, and Björn Hartmann. Collaboratively crowdsourcing workflows with turkomatic. In *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work - CSCW '12*, page 1003, New York, New York, USA, 2012. ACM Press. ISBN 9781450310864. doi: 10.1145/2145204.2145354. URL http://dl.acm.org/citation.cfm?doid=2145204.2145354.

[115] Walter S. Lasecki, Juho Kim, Nick Rafter, Onkur Sen, Jeffrey P. Bigham, and Michael S. Bernstein. Apparition: Crowdsourced user interfaces that come to life as you sketch them. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, pages 1925–1934, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3145-6. doi: 10.1145/2702123.2702565. URL http://doi.acm.org/10.1145/2702123.2702565.

[116] Jean Lave and Etienne Wenger. *Situated learning: Legitimate peripheral participation*. Cambridge university press, 1991.

[117] Dave Lee. Boston bombing: How internet detectives got it very wrong. *BBC News*, 19, 2013.

[118] Sang Won Lee, Yujin Zhang, Isabelle Wong, Yiwei Yang, Stephanie D. O'Keefe, and Walter S. Lasecki. Sketchexpress: Remixing animations for more effective crowd-powered prototyping of interactive interfaces. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, UIST '17, pages 817–828, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-4981-9. doi: 10.1145/3126594.3126595. URL http://doi.acm.org/10.1145/3126594.3126595.

[119] Tianyi Li, Kurt Luther, and Chris North. Crowdia: Solving mysteries with crowd-sourced sensemaking. *Proc. ACM Hum.-Comput. Interact.*, 2(CSCW):105:1–105:29, November 2018. ISSN 2573-0142. doi: 10.1145/3274374. URL http://doi.acm.org/10.1145/3274374.

[120] Tianyi Li, Kurt Luther, and Chris North. Crowdia: Solving mysteries with crowd-sourced sensemaking. *Proc. ACM Hum.-Comput. Interact.*, 2(CSCW), November 2018. doi: 10.1145/3274374. URL https://doi.org/10.1145/3274374.

[121] Tianyi Li, Asmita Shah, Kurt Luther, and Chris North. Crowdsourcing Intelligence Analysis with Context Slices. 2018.

[122] Tianyi Li, Asmita Shah, Kurt Luther, and Chris North. Crowdsourcing intelligence analysis with context slices. In *Chi'18 Sensemaking Workshop*, 2018.

[123] Tianyi Li, Chandler J. Manns, Chris North, and Kurt Luther. Dropping the baton? understanding errors and bottlenecks in a crowdsourced sensemaking pipeline. *Proc. ACM Hum.-Comput. Interact.*, 3(CSCW), November 2019. doi: 10.1145/3359238. URL https://doi.org/10.1145/3359238.

[124] Raanan Lipshitz, Neta Ron, and Micha Popper. Retrospective sensemaking and foresight: studying the past to prepare for the future. *Managing the future*, page 98, 2004.

[125] Greg Little, Lydia B Chilton, Max Goldman, and Robert C Miller. TurKit: Human Computation Algorithms on Mechanical Turk. URL http://delivery.acm.org/10.1145/1870000/1866040/p57-little.pdf?ip=45.3.66.147&id=1866040&acc=ACTIVE%20SERVICE&key=B33240AC40EC9E30%2E80AE0C8B3B97B250%2E4D4702B0C3E38B35%2E4D4702B0C3E38B35&CFID=992997097&CFTOKEN=20586175&__acm__=1507680503_716cb18f84c3885c.

[126] Greg Little, Lydia B Chilton, Max Goldman, and Robert C Miller. Exploring iterative and parallel human computation processes. In *Proceedings of the ACM SIGKDD workshop on human computation*, pages 68–76. ACM, 2010. doi: 10.1145/1837885.1837907.

[127] Greg Little, Lydia B Chilton, Max Goldman, and Robert C Miller. Exploring iterative and parallel human computation processes. In *Proceedings of the ACM SIGKDD workshop on human computation*, pages 68–76, 2010.

[128] Angli Liu, Stephen Soderland, Jonathan Bragg, Christopher H Lin, Xiao Ling, and Daniel S Weld. Effective crowd annotation for relation extraction. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 897–906, 2016.

[129] Kurt Luther, Scott Counts, Kristin B. Stecher, Aaron Hoff, and Paul Johns. Pathfinder: an online collaboration environment for citizen scientists. In *Proceedings of the 27th international conference on Human factors in computing systems*,

pages 239–248, Boston, MA, USA, 2009. ACM. ISBN 978-1-60558-246-7. doi: 10.1145/1518701.1518741. URL http://portal.acm.org/citation.cfm?id=1518741.

[130] Kurt Luther, Nathan Hahn, Steven P Dow, and Aniket Kittur. Crowdlines: Supporting synthesis of diverse information sources through crowdsourced outlines. In *Third AAAI Conference on Human Computation and Crowdsourcing*, 2015.

[131] Kurt Luther, Jari-Lee Tolentino, Wei Wu, Amy Pavel, Brian P. Bailey, Maneesh Agrawala, Björn Hartmann, and Steven P. Dow. Structuring, aggregating, and evaluating crowdsourced design critique. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work Social Computing*, CSCW '15, page 473–485, New York, NY, USA, 2015. Association for Computing Machinery. ISBN 9781450329224. doi: 10.1145/2675133.2675283. URL https://doi.org/10.1145/2675133.2675283.

[132] Kurt Luther, Jari-Lee Tolentino, Wei Wu, Amy Pavel, Brian P Bailey, Maneesh Agrawala, Björn Hartmann, and Steven P Dow. Structuring, aggregating, and evaluating crowdsourced design critique. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*, pages 473–485. ACM, 2015. doi: 10.1145/2675133.2675283.

[133] Kurt Luther, Jari-Lee Tolentino, Wei Wu, Amy Pavel, Brian P. Bailey, Maneesh Agrawala, Björn Hartmann, and Steven P. Dow. Structuring, aggregating, and evaluating crowdsourced design critique. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*, CSCW '15, pages 473–485, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-2922-4. doi: 10.1145/2675133.2675283. URL http://doi.acm.org/10.1145/2675133.2675283.

[134] Thomas W. Malone, Jeffrey V. Nickerson, Robert J. Laubacher, Laur Hesse Fisher, Patrick de Boer, Yue Han, and W. Ben Towne. Putting the Pieces Back Together

Again. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing - CSCW '17*, number February, pages 1661–1674, New York, New York, USA, 2017. ACM Press. ISBN 9781450343350. doi: 10.1145/2998181.2998343. URL http://dl.acm.org/citation.cfm?doid=2998181.2998343.

[135] Lena Mamykina, Bella Manoim, Manas Mittal, George Hripcsak, and Björn Hartmann. Design lessons from the fastest q&a site in the west. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 2857–2866. ACM, 2011.

[136] VK Chaithanya Manam and Alexander J Quinn. Wingit: Efficient refinement of unclear task instructions. In *Sixth AAAI Conference on Human Computation and Crowdsourcing*, 2018.

[137] Adam Marcus and Aditya Parameswaran. Crowdsourced Data Management: Industry and Academic Perspectives. *Foundations and Trends in Databases*, 6(1-2): 1–161, December 2015. ISSN 1931-7883, 1931-7891. doi: 10.1561/1900000044. URL https://www.nowpublishers.com/article/Details/DBS-044.

[138] Winter Mason and Duncan J. Watts. Financial incentives and the "performance of crowds". In *Proceedings of the ACM SIGKDD Workshop on Human Computation*, HCOMP '09, pages 77–85, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-672-4. doi: 10.1145/1600150.1600175. URL http://doi.acm.org/10.1145/1600150.1600175.

[139] Filippo Menczer. The spread of misinformation in social media. In *Proceedings of the 25th International Conference Companion on World Wide Web*, WWW '16 Companion, pages 717–717, Republic and Canton of Geneva, Switzerland, 2016. International World Wide Web Conferences Steering Committee. ISBN 978-1-4503-4144-8. doi: 10.1145/2872518.2890092. URL https://doi.org/10.1145/2872518.2890092.

[140] Vikram Mohanty, David Thames, and Kurt Luther. Photo sleuth: Combining collective intelligence and computer vision to identify historical portraits. In *ACM Conference on Collective Intelligence (CI 2018)*, 2018.

[141] Vikram Mohanty, Kareem Abdol-Hamid, Courtney Ebersohl, and Kurt Luther. Second opinion: Supporting last-mile person identification with crowdsourcing and face recognition. In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, volume 7, pages 86–96, 2019.

[142] Meredith Ringel Morris, Jeffrey P. Bigham, Robin Brewer, Jonathan Bragg, Anand Kulkarni, Jessie Li, and Saiph Savage. Subcontracting microwork. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, pages 1867–1876, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-4655-9. doi: 10.1145/3025453.3025687. URL http://doi.acm.org/10.1145/3025453.3025687.

[143] Meredith Ringel Morris, Jeffrey P Bigham, Robin Brewer, Jonathan Bragg, Anand Kulkarni, Jessie Li, and Saiph Savage. Subcontracting microwork. In *Proceedings of the 2017 CHI conference on human factors in computing systems*, pages 1867–1876. ACM, 2017.

[144] Lev Muchnik, Sinan Aral, and Sean J. Taylor. Social influence bias: A randomized experiment. *Science*, 341(6146):647–651, 2013. ISSN 0036-8075. doi: 10.1126/science.1240466. URL http://science.sciencemag.org/content/341/6146/647.

[145] Sean Collins Murray. Groupthink: Psychological studies of policy decisions and fiascoes, 1983.

[146] Johnny Nhan, Laura Huey, and Ryan Broll. Digilantism: An analysis of crowdsourcing and the boston marathon bombings. *The British Journal of Criminology*, 57(2):341–361, 2017. doi: 10.1093/bjc/azv118. URL http://dx.doi.org/10.1093/bjc/azv118.

[147] Stefanie Nowak and Stefan Rüger. How reliable are annotations via crowdsourcing: A study about inter-annotator agreement for multi-label image annotation. In *Proceedings of the International Conference on Multimedia Information Retrieval*, MIR '10, page 557–566, New York, NY, USA, 2010. Association for Computing Machinery. ISBN 9781605588155. doi: 10.1145/1743384.1743478. URL https://doi.org/10.1145/1743384.1743478.

[148] David Oleson, Alexander Sorokin, Greg Laughlin, Vaughn Hester, John Le, and Lukas Biewald. Programmatic gold: Targeted and scalable quality assurance in crowdsourcing. In *Workshops at the Twenty-Fifth AAAI Conference on Artificial Intelligence*, 2011.

[149] Daniel M. Oppenheimer, Tom Meyvis, and Nicolas Davidenko. Instructional manipulation checks: Detecting satisficing to increase statistical power. *Journal of Experimental Social Psychology*, 45(4):867 – 872, 2009. ISSN 0022-1031. doi: https://doi.org/10.1016/j.jesp.2009.03.009. URL http://www.sciencedirect.com/science/article/pii/S0022103109000766.

[150] Susannah BF Paletz, Joel Chan, and Christian D Schunn. Uncovering uncertainty through disagreement. *Applied Cognitive Psychology*, 30(3):387–400, 2016.

[151] Vineet Pandey, Amnon Amir, Justine Debelius, Embriette R. Hyde, Tomasz Kosciolek, Rob Knight, and Scott Klemmer. Gut Instinct: Creating Scientific Theories with Online Learners. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, pages 6825–6836, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-4655-9. doi: 10.1145/3025453.3025769. URL http://doi.acm.org/10.1145/3025453.3025769.

[152] Devi Parikh and C. Lawrence Zitnick. Human-Debugging of Machines. In *Neural*

Information Processing Systems, pages 1–5, 2011. URL https://filebox.ece.vt.edu/~parikh/human_debugging/.

[153] Peter Pirolli and Stuart Card. The sensemaking process and leverage points for analyst technology as identified through cognitive task analysis. volume 2005, pages 2–4, 2005. ISBN 0029-7844 (Print). doi: 10.1007/s13398-014-0173-7.2. URL http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:The+Sensemaking+Process+and+Leverage+Points+for+Analyst+Technology+as+Identified+Through+Cognitive+Task+Analysis#0%0Ahttps://www.e-education.psu.edu/geog885/sites/www.e-education.psu.edu.geog88.

[154] Peter Pirolli and Stuart Card. The sensemaking process and leverage points for analyst technology as identified through cognitive task analysis. In *Proceedings of international conference on intelligence analysis*, volume 5, pages 2–4. McLean, VA, USA, 2005.

[155] Daniela Retelny, Michael S. Bernstein, and Melissa A. Valentine. No workflow can ever be enough: How crowdsourcing workflows constrain complex work. *Proc. ACM Hum.-Comput. Interact.*, 1(CSCW):89:1–89:23, December 2017. ISSN 2573-0142. doi: 10.1145/3134724. URL http://doi.acm.org/10.1145/3134724.

[156] Daniela Retelny, Michael S Bernstein, and Melissa A Valentine. No workflow can ever be enough: How crowdsourcing workflows constrain complex work. *Proceedings of the ACM on Human-Computer Interaction*, 1(CSCW):1–23, 2017.

[157] Jeffrey Rzeszotarski and Aniket Kittur. CrowdScape: interactively visualizing user behavior and output. page 55, New York, New York, USA, 2012. ACM Press. ISBN 9781450315807. doi: 10.1145/2380116.2380125. URL http://dl.acm.org/citation.cfm?doid=2380116.2380125.

[158] Jeffrey Rzeszotarski and Aniket Kittur. Crowdscape: interactively visualizing user behavior and output. In *Proceedings of the 25th annual ACM symposium on User interface software and technology*, pages 55–62, 2012.

[159] Jeffrey M. Rzeszotarski and Aniket Kittur. Instrumenting the crowd: Using implicit behavioral measures to predict task performance. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, UIST '11, page 13–22, New York, NY, USA, 2011. Association for Computing Machinery. ISBN 9781450307161. doi: 10.1145/2047196.2047199. URL https://doi.org/10.1145/2047196.2047199.

[160] Jeffrey M Rzeszotarski and Aniket Kittur. Instrumenting the crowd: using implicit behavioral measures to predict task performance. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, pages 13–22. ACM, 2011.

[161] Nikhil Sharma. Sensemaking handoff: Theory and recommendations. In *CHI '07 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '07, page 1673–1676, New York, NY, USA, 2007. Association for Computing Machinery. ISBN 9781595936424. doi: 10.1145/1240866.1240880. URL https://doi.org/10.1145/1240866.1240880.

[162] Nikhil Sharma. Sensemaking handoff: When and how? volume 45, pages 1–12. Wiley-Blackwell, 6 2009. doi: 10.1002/meet.2008.1450450234. URL http://doi.wiley.com/10.1002/meet.2008.1450450234.

[163] Nikhil Sharma and George Furnas. Artifact usefulness and usage in sensemaking handoffs. volume 46, pages 1–19. Wiley-Blackwell, 2009. doi: 10.1002/meet.2009.1450460219. URL http://doi.wiley.com/10.1002/meet.2009.1450460219.

[164] Stephen Soderland. Learning Information Extraction Rules for Semi-Structured and

Free Text. volume 34, pages 233–272. Kluwer Academic Publishers, 1999. doi: 10.1023/A:1007562322031. URL http://link.springer.com/10.1023/A:1007562322031.

[165] John Stasko, Carsten Görg, and Zhicheng Liu. Jigsaw: Supporting Investigative Analysis through Interactive Visualization. volume 7, pages 118–132, 6 2008. doi: 10.1057/palgrave.ivs.9500180. URL http://dx.doi.org/10.1057/palgrave.ivs.9500180.

[166] Garold Stasser and William Titus. Pooling of unshared information in group decision making: Biased information sampling during discussion. *Journal of personality and social psychology*, 48(6):1467, 1985.

[167] Margaret-Anne Storey, Christoph Treude, Arie van Deursen, and Li-Te Cheng. The impact of social media on software engineering practices and tools. In *Proceedings of the FSE/SDP Workshop on Future of Software Engineering Research*, FoSER '10, pages 359–364, New York, NY, USA, 2010. ACM. ISBN 978-1-4503-0427-6. doi: 10.1145/1882362.1882435. URL http://doi.acm.org/10.1145/1882362.1882435.

[168] Maoyuan Sun, Peng Mi, Chris North, and Naren Ramakrishnan. Biset: Semantic edge bundling with biclusters for sensemaking. *IEEE transactions on visualization and computer graphics*, 22(1):310–319, 2015.

[169] Maoyuan Sun, Peng Mi, Chris North, and Naren Ramakrishnan. BiSet: Semantic Edge Bundling with Biclusters for Sensemaking. volume 22, pages 310–319, 1 2016. doi: 10.1109/TVCG.2015.2467813. URL http://ieeexplore.ieee.org/document/7192715/.

[170] Yla Tausczik and Mark Boons. Distributed Knowledge in Crowds: Crowd Performance on Hidden Profile Tasks. In *Twelfth International AAAI Conference on Web and Social Media*, June 2018. URL https://aaai.org/ocs/index.php/ICWSM/ICWSM18/paper/view/17817.

[171] Yla Tausczik and Mark Boons. Distributed knowledge in crowds: Crowd performance on hidden profile tasks. In *Twelfth International AAAI Conference on Web and Social Media*, 2018.

[172] Yi Tay, Luu Anh Tuan, and Siu Cheung Hui. Multi-range reasoning for machine comprehension. *arXiv preprint arXiv:1803.09074*, 2018.

[173] Alice Toniolo, Timothy J. Norman, Anthony Etuk, Robin Wentao Ouyang, Nir Oren, Timothy Dropps, John A. Allen, Federico Cerutti, Robin Wentao Ouyang, Mani Srivastava, Nir Oren, Timothy Dropps, John A. Allen, and Paul Sullivan. Supporting Reasoning with Different Types of Evidence in Intelligence Analysis. In *AAMAS '15 Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, AAMAS '15, pages 781–789, Richland, SC, 2015. International Foundation for Autonomous Agents and Multiagent Systems. ISBN 978-1-4503-3413-6. URL http://dl.acm.org/citation.cfm?id=2772879.2773254http://dl.acm.org/ft_gateway.cfm?id=2773254&type=pdf.

[174] John W Tukey. We need both exploratory and confirmatory. *The American Statistician*, 34(1):23–25, 1980.

[175] Melissa A. Valentine, Daniela Retelny, Alexandra To, Negar Rahmati, Tulsee Doshi, and Michael S. Bernstein. Flash Organizations: Crowdsourcing Complex Work by Structuring: Crowds As Organizations. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems - CHI '17*, pages 3523–3537, New York, New York, USA, 2017. ACM Press. ISBN 9781450346559. doi: 10.1145/3025453.3025811. URL http://dl.acm.org/citation.cfm?doid=3025453.3025811.

[176] Melissa A. Valentine, Daniela Retelny, Alexandra To, Negar Rahmati, Tulsee Doshi, and Michael S. Bernstein. Flash organizations: Crowdsourcing complex work by

structuring crowds as organizations. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, pages 3523–3537, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-4655-9. doi: 10.1145/3025453.3025811. URL http://doi.acm.org/10.1145/3025453.3025811.

[177] Vasilis Verroios and Michael S Bernstein. Context Trees: Crowdsourcing Global Understanding from Local Views. Number 1351131, pages 210–219, 2014. URL http://ilpubs.stanford.edu:8090/1105/.

[178] Vasilis Verroios and Michael S Bernstein. Context trees: Crowdsourcing global understanding from local views. In *Second AAAI Conference on Human Computation and Crowdsourcing*, 2014.

[179] Luis von Ahn and Laura Dabbish. Labeling images with a computer game. pages 319–326, New York, New York, USA, 2004. ACM Press. ISBN 1581137028. doi: 10.1145/985692.985733. URL http://portal.acm.org/citation.cfm?doid=985692.985733.

[180] Hanna M. Wallach and Hanna M. Topic modeling. In *Proceedings of the 23rd international conference on Machine learning - ICML '06*, pages 977–984, New York, New York, USA, 2006. ACM Press. ISBN 1595933832. doi: 10.1145/1143844.1143967. URL http://portal.acm.org/citation.cfm?doid=1143844.1143967.

[181] Jiannan Wang, Tim Kraska, Michael J. Franklin, and Jianhua Feng. Crowder: Crowdsourcing entity resolution. *Proc. VLDB Endow.*, 5(11):1483–1494, July 2012. ISSN 2150-8097. doi: 10.14778/2350229.2350263. URL https://doi.org/10.14778/2350229.2350263.

[182] Jiannan Wang, Tim Kraska, Michael J. Franklin, and Jianhua Feng. CrowdER: Crowdsourcing Entity Resolution. pages 1483–1494, 2012. ISBN 2150-8097. doi: 10.14778/2350229.2350263. URL http://arxiv.org/abs/1208.1927.

[183] Daniel S Weld, Christopher H Lin, and Jonathan Bragg. Artificial intelligence and collective intelligence. *Handbook of Collective Intelligence*, pages 89–114, 2015.

[184] Ashley Wheat, Simon Attfield, and Bob Fields. Developing a Model of Distributed Sensemaking: A Case Study of Military Analysis. volume 3, page 1, 2 2016. doi: 10.3390/informatics3010001. URL http://www.mdpi.com/2227-9709/3/1/1http://www.mdpi.com/2227-9709/3/1/1/htmhttp://www.mdpi.com/2227-9709/3/1/1/pdf.

[185] Mark E Whiting, Dilrukshi Gamage, Snehalkumar S Gaikwad, Aaron Gilbee, Shirish Goyal, Alipta Ballav, Dinesh Majeti, Nalin Chhibber, Angela Richmond-Fuller, Freddie Vargus, et al. Crowd guilds: Worker-led reputation and feedback on crowdsourcing platforms. 2016. doi: 10.1145/2998181.2998234.

[186] Wesley Willett, Jeffrey Heer, Joseph Hellerstein, and Maneesh Agrawala. Commentspace: structured support for collaborative visual analysis. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, pages 3131–3140. ACM, 2011. doi: 10.1145/1978942.1979407.

[187] Wesley Willett, Jeffrey Heer, and Maneesh Agrawala. Strategies for crowdsourcing social data analysis. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 227–236, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1015-4. doi: 10.1145/2207676.2207709. URL http://doi.acm.org/10.1145/2207676.2207709.

[188] William Wright, David Schroh, Pascale Proulx, Alex Skaburskis, and Brian Cort. The sandbox for analysis. In *Proceedings of the SIGCHI conference on Human Factors in computing systems - CHI '06*, page 801, New York, New York, USA, 2006. ACM

Press. ISBN 1595933727. doi: 10.1145/1124772.1124890. URL http://portal.acm.org/citation.cfm?doid=1124772.1124890.

[189] William Wright, David Schroh, Pascale Proulx, Alex Skaburskis, and Brian Cort. The sandbox for analysis: Concepts and methods. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '06, pages 801–810, New York, NY, USA, 2006. ACM. ISBN 1-59593-372-7. doi: 10.1145/1124772.1124890. URL http://doi.acm.org/10.1145/1124772.1124890.

[190] Hao Wu, Michael Mampaey, Nikolaj Tatti, Jilles Vreeken, M Shahriar Hossain, and Naren Ramakrishnan. Where do i start?: algorithmic strategies to guide intelligence analysts. In *Proceedings of the ACM SIGKDD Workshop on Intelligence and Security Informatics*, page 3. ACM, 2012. doi: 10.1145/2331791.2331794.

[191] Meng-Han Wu and Alexander James Quinn. Confusing the crowd: Task instruction quality on amazon mechanical turk. In *Fifth AAAI Conference on Human Computation and Crowdsourcing*, 2017.

[192] H. Xie and J. C. S. Lui. Incentive mechanism and rating system design for crowdsourcing systems: Analysis, tradeoffs and inference. *IEEE Transactions on Services Computing*, 11(1):90–102, Jan 2018. ISSN 1939-1374. doi: 10.1109/TSC.2016.2539954.

[193] Anbang Xu, Shih-Wen Huang, and Brian Bailey. Voyant: Generating Structured Feedback on Visual Designs Using a Crowd of Non-Experts. pages 1433–1444, 2014. ISBN 9781450325400. doi: 10.1145/2531602.2531604. URL http://dl.acm.org/citation.cfm?id=2531602.2531604.

[194] Roman V Yampolskiy. Turing test as a defining feature of ai-completeness. In *Artificial intelligence, evolutionary computing and metaheuristics*, pages 3–17. Springer, 2013. doi: 10.1007/978-3-642-29694-9_1.

[195] Haoqi Zhang, Edith Law, Rob Miller, Krzysztof Gajos, David Parkes, and Eric Horvitz. Human computation tasks with global constraints. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, page 217–226, New York, NY, USA, 2012. Association for Computing Machinery. ISBN 9781450310154. doi: 10.1145/2207676.2207708. URL https://doi.org/10.1145/2207676.2207708.

[196] Haoqi Zhang, Edith Law, Rob Miller, Krzysztof Gajos, David Parkes, and Eric Horvitz. Human computation tasks with global constraints. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 217–226, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1015-4. doi: 10.1145/2207676.2207708. URL http://doi.acm.org/10.1145/2207676.2207708.

[197] Haoqi Zhang, Edith Law, Rob Miller, Krzysztof Gajos, David Parkes, and Eric Horvitz. Human computation tasks with global constraints. In *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems*, CHI '12, pages 217–226, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1015-4. doi: 10.1145/2207676.2207708. URL http://doi.acm.org/10.1145/2207676.2207708.

[198] Jian Zhao, Michael Glueck, Petra Isenberg, Fanny Chevalier, and Azam Khan. Supporting Handoff in Asynchronous Collaborative Sensemaking Using Knowledge-Transfer Graphs. In *IEEE Transactions on Visualization and Computer Graphics*, volume 24, pages 1–1, 1 2017. doi: 10.1109/TVCG.2017.2745279. URL http://ieeexplore.ieee.org/document/8017596/.

[199] Jian Zhao, Michael Glueck, Petra Isenberg, Fanny Chevalier, and Azam Khan. Supporting handoff in asynchronous collaborative sensemaking using knowledge-transfer graphs. *IEEE transactions on visualization and computer graphics*, 24(1):340–350, 2017.

[200] Haiyi Zhu, Steven P. Dow, Robert E. Kraut, and Aniket Kittur. Reviewing versus doing. In *Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing - CSCW '14*, pages 1445–1455, New York, New York, USA, 2014. ACM Press. ISBN 9781450325400. doi: 10.1145/2531602.2531718. URL http://dl.acm.org/citation.cfm?doid=2531602.2531718.

Appendices

Appendix A

Appendix for Chapter 4

A.1 Datasets

Given the complexity of the entire sensemaking process, we evaluate our pipeline with simplified datasets in three levels of difficulty. Dataset with more documents, more elements (who, what, where, when) and more complicated relationships among elements are considered more difficult.

The easy dataset is adapted from one of the brain teasers from Braingle [51], the answer being *Serina is the culprit*. There are two relevant documents: Document 1 introduces the background setting and Document 2 lays out the suspects and information about them. We added a third irrelevant document as noise, and masked the original names to prevent crowds from finding the solution online. There are 227 words in all three documents, with 162 words in the two key documents (Table A.1).

The moderate dataset is adapted from the popular board game Clue, where there is a limited number of suspects (who), weapons (what), locations (where), and one known murder time (when). We picked three suspects, two weapons and three locations in the whole dataset (Table A.2), and the correct answer being *Miss Scarlett killed Mr. Boddy (victim) in his kitchen with a knife, because she will be bequeathed with the large estate after his death.*

The difficult dataset is part of the *Sign of Crescent* dataset [89]. It is used as a training

Document 1	**Document 2**	**Document 3**
One hot, dry day Neva saw Mr. Potter shaking his head as he stood by his flowerbed. "Somebody ruined all my flowers," he said. "I had the hose out watering them. When I went to put it away, somebody tromped through the rows and stomped on my flowers." "Who'd do a mean thing like that?" Neva asked. Mr. Potter sighed. "Somebody who likes mischief, I guess." "I'm walking to the mall now. Maybe later I can find out who did it," Neva told him.	On Neva's way to the mall, she saw three girls playing hopscotch. She decided to stop and watch how expertly they moved over the chalk marks. Lucy had to hop very carefully because one of the straps was broken on her left sandal. Cathy hopped slowly. She wore purple sneakers that looked worn-out. Cathy seemed worn-out, too. Serina hopped the fastest. The muddy soles of her white jogging shoes hardly seemed to touch the sidewalk as she moved.	Ada Peterson is a graduate student in ABC Tech. She went on a vacation to Yellowstone National Park this August with her family. She spent 3 days at her cousin Elaine's house in Los Angeles before that. She spent a week there, and before she came back to school on Aug 28th, she went to Utah for 2 days to visit her old friend Cindy.

Table A.1: Easy dataset adapted from a brain teaser. The correct answer is that Serina is the culprit.

material for intelligence analysts. There are 41 report documents regarding three fictional terrorist attacks. Each plot involves a group of at least four suspicious people. And each report document contains a single prose paragraph of 33 to 210 words (Fig. A.3). We took nine of the documents that contain evidence of one of the attack plots: *A C-4 plastic explosive bomb, will be detonated at 0900hrs on 30 April,2003, by Hamid Alwan [alias Mark Davis] in New York Stock Exchange. Support Hamid with money and bomb storage and transportation is a group of terrorists: Muhammed bin Harazi [alias Abdul Ramazi], Hani al Halak, Sahim Albakri [alias Bagwant Dhaliwal].* We added two irrelevant documents as noise.

Evidence	**Miss Scarlett***	**Prof. Plum**	**Reverend Green**
Means	has knife	handgun (wrong weapon)	has knife
Motive	inherit victim's property on his death	wife has affair with victim	victim stole money from their business
Opportunity	visited victim's house	car driving past (wrong location)	phone call (wrong location)

Table A.2: Moderate dataset skeleton adapted from the card game Clue.

Report Date 14 April, 2003. CIA: From an interrogation of a cooperative detainee in Guantanamo. Detainee says he trained daily with a man named Ziad al Shibh at an Al Qaeda explosives training facility in the Sudan in 1994. From a captured laptop computer in Afghanistan it is learned that Ziad al Shibh holds a United Arab Emirates passport in the name Faysal Goba. INS check reveals that a Faysal Goba, from the United Arab Emirates, entered the USA on a travel visa in January of 2003 stating that he would be visiting a person named Clark Webster in Richmond, Va. The contact address given by Goba was: 1631 Capitol Ave., Richmond VA; phone number: 804-759-6302.
Report Date 27 April, 2003. Intercept of cell phone 804-774-8920. In a very brief call from this number to phone number 703-659-2317 on 26 April, 2003, the caller speaks in Arabic. A translation reads: "We are now prepared to take the crescent to victory".

Table A.3: Example documents from the difficult dataset adapted from *The Sign of the Crescent*.

A.2 Crowd Analysis of Easy Dataset

Step 1: The crowds successfully identified that Document 1 and Document 2 are relevant.

Step 2: The information pieces extracted by the crowds are:

1. Neva witness Mr. Potter shaking his head by the flowerpots.
2. Mr. Potter claimed someone ruined his flowers by stomping on them after he watered them.
3. Neva wondered who it was.
4. Mr. Potter didn't know but guessed it was for mischief's sake.
5. Neva said he might help figure it out later.
6. On that day, Neva saw three girls playing hopscotch and watched them.
7. Lucy had a broken strap on her left sandal.
8. Cathy looked worn out and her shoes were worn too.
9. Serina was fast but had muddy shoes.
10. Any of these three girls could be the culprit.

Step 3: The schemas organized by crowds are:

Group 1: Possible suspects

6. On that day, Neva saw three girls playing hopscotch and watched them.
7. Lucy had a broken strap on her left sandal.
8. Cathy looked worn out and her shoes were worn too.

9. Serina was fast but had muddy shoes.
10. Any of these three girls could be the culprit.

Group 2: Evidence Mr. Potter's flowerpots were destroyed

1. Neva witness Mr. Potter shaking his head by the flowerpots.
2. Mr. Potter claimed someone ruined his flowers by stomping on them after he watered them.

Group 3: Reflections

3. Neva wondered who it was.
4. Mr. Potter didn't know but guessed it was for mischief's sake.
5. Neva said he might help figure it out later.

Group 4: Could be Neva

4. Neva said he might help figure it out later.

Step 4: The hypotheses developed in Step 4 are:

Hypothesis 1, Group 1, 2, & 3: Mr. Potter's flowerpots were destroyed and claimed someone ruined his flowers by stomping them. Neva saw three girls playing hopscotch and noticed that one girl, Lucy had a broken strap on her sandal. It is likely that Lucy stomped on the flowers and in doing so her strap on her left sandal caught on something and broke, or from the force or her action that is how the flowers were ruined.

Hypothesis 2, Group 1, 2, & 3: Mr. Potter's flowerpots were destroyed and claimed someone ruined his flowers by stomping them. Neva saw three girls playing hopscotch and noticed

that one girl, Cathy looked exhausted and had worn out shoes. Cathy looked so worn out due to the fact that she ruined Mr. Potter's flowers and ran off in a rush as to not get caught

Hypothesis 3, Group 1, 2, & 3: Mr. Potter's flowerpots were destroyed and claimed someone ruined his flowers by stomping them. Neva saw three girls playing hopscotch and noticed that one girl, Serina was very agile in playing hopscotch and had muddy shoes. Since Mr. Potter had just watered his flowers the soil would have been moist/damp and it makes sense that Serina ruined his flowers and ran back to her friends to play with them which is why her shoes are muddy

Step 5: The final presentation by crowds is:

I think it was Serina who had the muddy shoes after playing hopscotch. Her shoes were muddy so that could indicate that she went into the just watered flower bed. Maybe she only ran through it to get to her friends so they could play but she might have stomped on the flowers on her way to the play area. She was in a hurry and not paying attention to what she was doing.

A.3 Crowd Analysis of Moderate Dataset

Step 1: The crowds identified all documents about Miss Scarlett, documents about Prof. Plum's motive and wrong location, and Reverend Green's means and motive as relevant, i.e. all relevant documents were retrieved, one irrelevant document was retrieved (precision=85.7%, recall=100%).

Step 2: The information pieces extracted by crowds are:

1. Miss Scarlett visited Mr. Boddy's house on the night of his death to return some books

2. Miss Scarlett will inherit Mr. Boddy's large estate in the event of his death as Mr. Boddy's niece and nearest living relative
3. Miss Scarlett's personal trainer Roger saw an ivory-handled fold-up knife in her gym bag a month ago, but she told police she has lost the knife several weeks ago.
4. Professor Plum didn't know that his wife Linda had been having an affair with Mr. Boddy
5. Professor Plum's car was seen driving past Mr. Boddy's house on the night of his death
6. Reverend Green called Mr. Boddy's house twice on the night of his death
7. Reverend Green kept a black utility knife in his pocket, but he told authorities that the knife had broken months ago and he had discarded it.
8. Reverend Green suspected Mr. Boddy had stolen money from the failed business they had together several years ago

Step 3: The information tagged by crowds can be used to generate the following profiles:
Step 4: The single elimination competition results by crowds are shown in Fig. A.1

Step 5: The final presentation created by the crowds is:

Miss Scarlet killed him. She was seen at Mr Boddy's house on the night of his death. She also had the murder weapon seen by her trainer in her bag. Also, she had the motive since she would inherit his estate

A.3.1 Additional Experiment Results of Moderate Dataset

Step 1: Same as the above described experiment.

	Miss Scarlett	Prof. Plum	Reverend Green	Roger
Means	Miss Scarlett's personal trainer Roger saw an ivory-handled fold-up knife in her gym bag a month ago, but she told police she has lost the knife several weeks ago.		Reverend Green kept a black utility knife in his pocket, but he told authorities that the knife had broken months ago and he had discarded it.	Miss Scarlett's personal trainer Roger saw an ivory-handled fold-up knife in her gym bag amonth ago, but she told police she has lost the knife several weeks ago.
Motive	Miss Scarlett will inherit Mr. Boddy's large estate in the event of his death as Mr. Boddy's niece and nearest living relative	Professor Plum didn't know that his wife Linda had been having an affair with Mr. Boddy	Reverend Green suspected Mr. Boddy had stolen money from the failed business they had together several years ago	
Opportunity	Miss Scarlett visited Mr. Boddy's house on the night of his death to return some books	Professor Plum's car was seen driving past Mr. Boddy's house on the night of his death		

Table A.4: Profiles generated from information pieces tagged by crowds in Step 3.

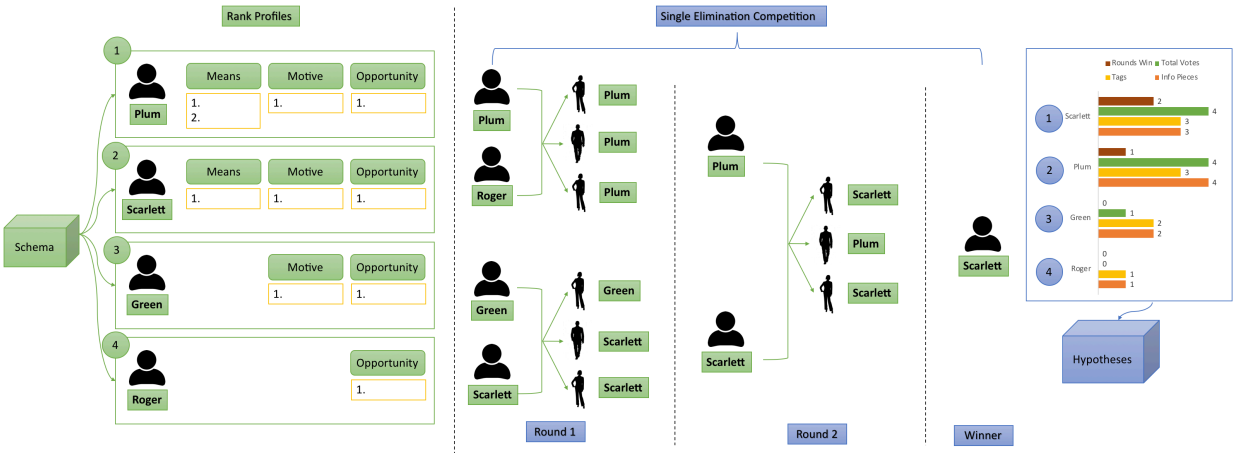

Figure A.1: Single elimination competition of profiles in Step 4.

Step 2: The information pieces extracted by crowds are:

1. Professor Plum has an active permit to carry a concealed handgun
2. Professor Plum most recently renewed the concealed carry permit two months ago.
3. Phone records show Reverend Green called Mr. Boddy's house twice on the night of his death
4. Sharon Miller possibly saw a car matching the description of Professor Plum's car driving past Mr. Boddy's house on the night of his death.
5. Sharon Miller is Mr. Boddy's neighbor.
6. Green kept a black utility knife in his pocket, for whittling wood carvings in his spare time.
7. Green told authorities that the knife had broken months ago, and he had discarded it.
8. Miss Scarlett was at Mr. Boddy's house the night he died.
9. Miss Scarlett said she was only returning books.

10. Miss Scarlett's personal trainer, Roger, told authorities that a month ago, he had seen an ivory-handled fold-up knife in Scarlett's gym bag
11. Ms. Scarlett lost the knife weeks ago.
12. Mr. Boddy has a large estate that will go to his nearest relative.
13. Mr. Boddy's nearest living relative was Miss Scarlett.
14. Professor Plum's wife, Linda was having an affair with Mr. Boddy.
15. Linda said she didn't believe that Professor Plum knew about the affair.
16. Reverend Green and Mr. Body were partners in a failed business
17. Reverend Green suspected Mr. Boddy of stealing money

Step 3: The information tagged by crowds generated five profiles. Following the order of amount of evidence: Scarlett, Green, Plum, Miller, and Linda (Table A.5).

Step 4: There were two rounds of competition. The first round knocked out Miller and Linda, and the second round knocked out Green and Plum.

Step 5: The final presentation created by the crowds is:

Scarlett was known to have a knife similar to the weapon used in the murder, she was at Mr Boddy's house and was also the last person who saw the victim alive.

A.4 Assumptions

To manage the scope of the problem, we enforce some assumptions on the initial data input and the final result output, focusing on the specific problem of text analysis we aim to tackle.

	Scarlett	Plum	Green	Miller	Linda
Means	Miss Scarlett's personal trainer, Roger, told authorities that a month ago, he had seen an ivory-handled fold-up knife in Scarlett's gym bag. Ms. Scarlett lost the knife weeks ago.	Professor Plum most recently renewed the concealed carry permit two months ago.	Green kept a black utility knife in his pocket, for whittling wood carvings in his spare time. Green told authorities that the knife had broken months ago, and he had discarded it.		
Motive	Mr. Boddy has a large estate that will go to his nearest relative. Mr. Boddy's nearest living relative was Miss Scarlett.	Professor Plum's wife, Linda was having an affair with Mr. Boddy. Linda said she didn't believe that Professor Plum knew about the affair.	Reverend Green and Mr. Body were partners in a failed business. Reverend Green suspected Mr. Boddy of stealing money		Professor Plum's wife, Linda was having an affair with Mr. Boddy.
Opportunity	Miss Scarlett was at Mr. Boddy's house the night he died. Miss Scarlett said she was only returning books.	Sharon Miller possibly saw a car matching the description of Professor Plum's car driving past Mr. Boddy's house on the night of his death.	Phone records show Reverend Green called Mr. Boddy's house twice on the night of his death	Sharon Miller possibly saw a car matching the description of Professor Plum's car driving past Mr. Boddy's house on the night of his death.	

Table A.5: Additional experiments: Profiles generated by information pieces tagged by crowds in Step 3.

The identification of data elements in each step of the pipeline will be defined based on these assumptions in following sections. We assume a crime solving or intelligence scenario.

A.4.1 Assumptions about External Data Resources

A1. There is a general investigation goal (global context). We assume there is a general investigation goal to guide the whole sensemaking process. For example, we know there is a murder case (known victim, time, and location), or we suspect there is a potential terrorist attack (undecided who, what, where, when).

A2. Source materials are narrative texts. Crime plots are diffused or obfuscated in the text with noise, in some latent structure. The documents are modularized in some uniform way, and can be disassembled into sentences, paragraphs or whole documents.

A3. Entities and their relationships are from the source texts. The key entities constructing the crime plots are all in the text, but relationships between them may be more or less explicit, which is why algorithms are not enough to uncover the hidden plots.

A4. No external information is required to solve the case. Common sense knowledge is enough to understand and analyse the source material. All necessary information is covered in the narrative texts and agents do not need to consult external information sources.

A5. Privacy and confidentiality is out of scope. Although we are using fictional crime-related evidence data as the example dataset, our main focus is the analysis of text data. For confidential datasets, the strategy could be applied within a private crowd (e.g. employees).

A.4.2 Assumptions about Reportable Results

A6. Assume an investigation report. Since the initial data input assumes crime plots are hidden in narrative texts, the final outputs should be an investigation result reportable to a

potential client.

A7. Event description fulfills formula for complete stories. The final results should conform to a simple template of a complete story comprised of: who, what (method of crime), where, and when with necessary supporting evidence.

A8. Each answer component has a finite number of options. There is a finite number of options for each of the four W's mentioned above, based on the content of the dataset.

A9. Missing links mean the solution is not correct. In terms of evaluation, if any of the W's are missing or any of the connections between entities are missing, the results are considered incomplete.

A10. Simpler explanations are preferred. In the final stage of analysis, among several candidate hypotheses with the same level of correctness, simpler candidates are preferred.

A11. Constraints are enforced by resources. There are limits like elapsed time, number of guesses allowed with given resources that constrains the analysis procedure. The pipeline cannot keep running forever and must stop when the limits are met.

Appendix B

Appendix for Chapter 5

B.1 Gold standard analysis and decision rationales

We describe the final gold standard output for each step and the decision rationales in the generating process.

10 relevant documents. We trace the source documents of each piece of key evidence used in the Wigmore chart and mark them as the *gold-standard relevant documents*.

19 most important info pieces. The answer sheet listed 20 key evidence parsed from the documents that contribute to solving the mystery. The key evidence includes both direct evidence and supporting clues, some are inferences that cannot be directly generated from one document only. In order to develop a baseline performance, we focus on only the direct evidence and assume the condition where each microtask only has access to one document. We break down the direct evidence used in the Wigmore chart into simpler sentences that 1) can be generated from one single document, 2) are structured as "who, what, where, when" as much as possible. This is to match the baseline condition of the step 2 microtasks where each worker only has access to one document. The resulting 19 info pieces are *gold-standard info pieces*.

5 location profiles. The answer sheet organizes the key evidence vertically by deductive reasoning, but not horizontally into profile schemata. We manually tag the single-document gold standard info pieces by whether they contain information about any known "*Terrorist*", the C-4 explosive "*Weapon*", and the planned attack "*Time*". Some information pieces need to be put together with others to reveal meaningful evidence; we mark these as hidden tags. For possible target locations, we extract all the locations mentioned in the information pieces. It's worth noting that the names and resolutions of the locations are tricky. For example, the correct answer "*NYSE (New York Stock Exchange)*" is also in New York City. Since both "NYC" and "New York City" appeared in the documents by themselves, we extracted New York City [NYC] as one of the possible target locations. Putting the info pieces about each location together is the *gold-standard location profiles*.

Likelihood ranking of profiles. The answer sheet also doesn't rank the likelihood of all locations mentioned in the dataset. We rank the locations by 1) their geographical distance to the real target location (NYSE), 2) the number of terrorist activities, and 3) the minimum depth of its mention in the Wigmore chart. We rank NYC, the lower resolution location containing the correct answer NYSE, both as the first place. The final ranking of likelihood is New York Stock Exchange [NYSE] = New York City [NYC] > Empire State Vending Services [ESVS] > carpet store in North Bergen, NJ > Springfield, VA.

Final presentation. The answer sheet contains a conclusion statement and almost two pages of an article detailing the process of terrorist coordinating the attack. The article connects the facts and develops hypotheses but also involves domain knowledge and speculative details (not mentioned nor derived from the given documents, e.g. "Several days before the delivery of the vending machine containing the bomb, Alwan goes to the NYSE to fill a coffee, tea, hot chocolate machine and, in the process, disrupts its functioning.")

Thus, we define the *gold-standard final presentation* not as a pragraph but regulates the most important facts to mention, namely:

- Joed Shearper is 1) a terrorist 2) with explosive training and 3) has access (to the vending machines in) the New York Stock Exchange under 4) the alias Devon Citopper.
- Cedric Whappadder 5) stores buckets of C-4 explosives in his (cover-up) carpet shop (in North Bergen, New Jersey).
- Joed Shearper (alias Devon Citopper) and Irving Sprunkiddle (alias Virgil Sneworf) are both terrorists. They live in the same apartment. One of them 6) picked up C-4 explosives from Cedric Whappadder's carpet shop.

Crowd generated presentation with gold-standard input. With the gold-standard profile, the crowd workers were able to reconstruct the crime from the given profile into a cohesive narrative. Below is the final presentation by the crowd, with all 6 matched facts from our gold-standard analysis boldfaced:

> The New York Stock Exchange (NYSE) may be under imminent threat of attack by several known terror suspects. One suspect, a Saudi **explosives expert** from named Joed Shearper (**aka "Devon Citopper"**), possesses a **NYSE** vendor's ID through his employment as a **vending machine operator**, providing the group both clearance and expertise for planning and conducting an attack. Alwan lives at the **same address** (2462 Myrtle Ave, Apt 307, Queens) and works at the same vending machine company as a known Pakistani **Taliban member**, Irving Sprunkiddle (aka "Virgil Sneworf"). A third suspect, **Cedric Whappadder**, as recently as April 26, 2003, **had access to C-4 explosives** that could be employed in a terror incident. Hallak manages a **carpet store** in North Bergen,

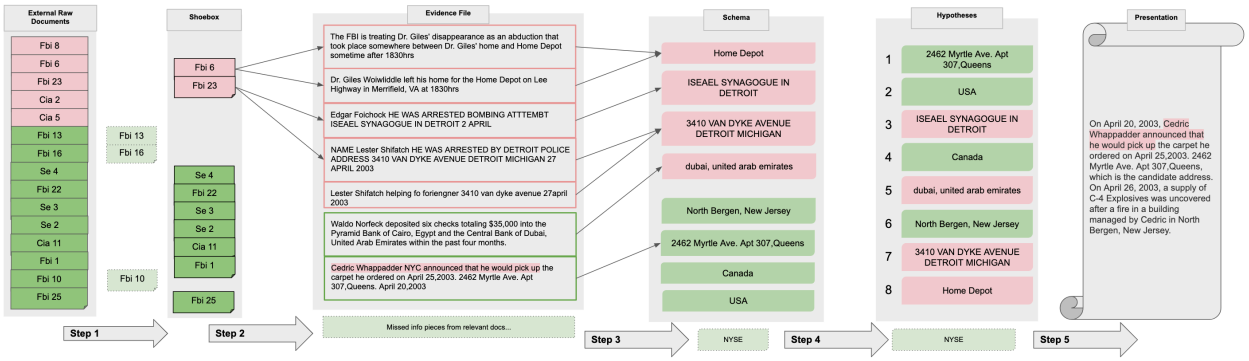

Figure B.1: Error Propagation in Uni-item Condition: the pink colored items are irrelevant documents, info pieces, and location profiles; the green colored ones are relevant to solving the mystery.

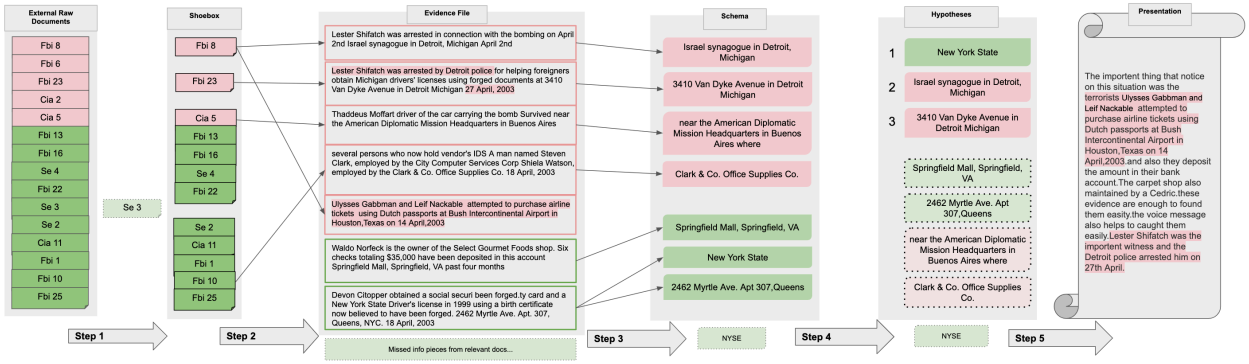

Figure B.2: Error propagation in triple-item condition

NJ; after several previous calls, on April 22 a caller from a number associated with Alwan and Albakri's Queen's address (718-352-8479) stated that he would **pick up a previously ordered carpet from the store's location**. Hallak has since vanished. Thus, this terror cell has the capability to attack the NYSE as believed.

B.2 Error propagation shown in diagrams

We present examples of error propagation in each step under the uni-item (Fig. B.1), triple-

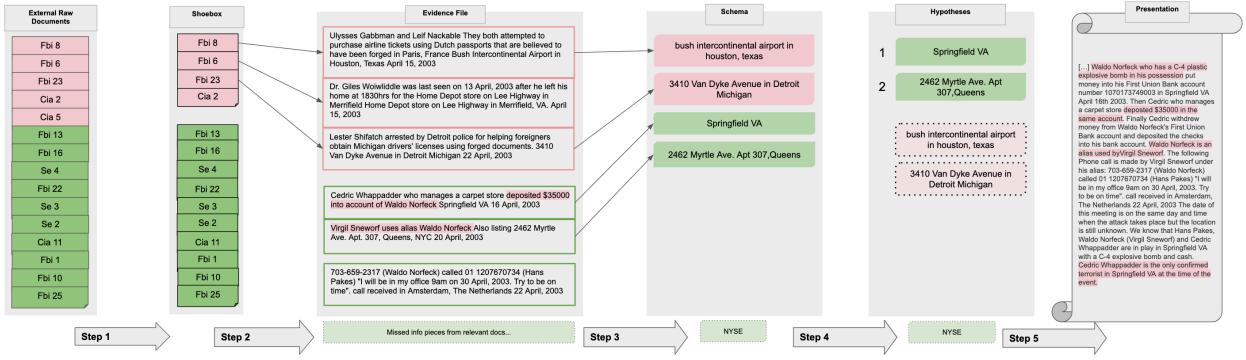

Figure B.3: Error propagation in all-item condition

item (Fig. B.2), and all-item (Fig. B.3).