Connect the Dots: Supporting Intelligence Analysis with Crowdsourcing, Context Slices, and Visualization

Author Name
Affiliation
Address
Country
Email

Author Name
Affiliation
Address
Country
Email

Author Name
Affiliation
Address
Country
Email

Author Name
Affiliation
Address
Country
Email

Author Name
Affiliation
Address
Country
Email

Author Name
Affiliation
Address
Country
Email

ABSTRACT

Making sense of large text datasets is a difficult problem in many domains and does not scale well for individuals. Crowdsourcing presents new opportunities for large-scale sensemaking, but we must first overcome the challenge of enabling many distributed novice workers to contribute meaningfully. In this paper, we explore the use of non-expert crowds to support expert analysts in complex sensemaking, focusing on the task of finding connections between entities in documents with narrative textual information. We introduce a novel concept called *context slices* in which datasets are restructured to enable in-depth inquiry by transient novice workers. We implemented this concept in a web application, *Connect the Dots*, in which crowds build subgraphs of entity relationships that can be layered and visualized for expert analysts. Our results suggest that with context slices, crowds are able to find most of the connections that analysts need, along with accurate and meaningful description labels, and can be used to retrieve and schematize information from the source documents.

Keywords

Crowdsourcing; Information Visualization; Intelligence Analysis; Sensemaking

1 INTRODUCTION

We are in the midst of a data deluge that challenges our traditional methods for making sense of the world. In the field of intelligence analysis, analysts must deal with a tremendous amount of rapidly changing data with diverse or unknown structures. Crowdsourcing presents new opportunities to manage this challenge by augmenting the cognitive work of individual analysts, providing richer analysis than automated approaches and scaling better than traditional intelligence work [6, 7, 18]. 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 [26].

To address this challenge, we introduce a novel concept called *context slices* in which datasets are restructured to enable in-depth inquiry by transient novice crowd workers on Amazon Mechanical Turk. Context slices decompose the dataset into smaller subsets that are appropriate for one crowd worker to analyze as a micro-task. In this work, we focus on supporting expert intelligence analysis of textual datasets containing numerous small narrative documents, with the help of crowdsourced micro-tasks for finding connections between entities in the documents. In this case, context slices are small groups of documents. We experiment with different slicing methods and explore to what extent crowds can find the connections needed by expert analysts, when only examining the data through context slices.

To explore the utility of context slices, we developed a web application, *Connect the Dots*, that helps crowd workers build subgraphs of entity relationships that can be layered and visualized for expert analysis. We conducted an experiment in which 275 paid crowd workers used this software to generate nearly 6000 pairs of entities and corresponding descriptions from documents about a fictional terrorist plot, simulating a real-world intelligence analysis task.

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

We also 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 also 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.

Our main contribution is exploration of 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.

2 RELATED WORK

2.1 Expert Sensemaking Practices

Making sense of large amounts of textual data can be a demanding cognitive task that is often performed by experts. In this section, we review prior research that studies and models the sensemaking process of individuals and groups of expert analysts.

Supporting expert intelligence analysis requires a clear understanding of the process and key components. Researchers have studied and modeled the sensemaking process of experts to enable better training and performance [27]. Two influential such models are the data-frame theory [21] and the sensemaking loop [26]. The data-frame theory describes the iterative process of analysts fitting external information in their mental model and updating their mental model to accommodate new knowledge. The sensemaking loop identifies multiple stages of intelligence analysis and organizes the process into two primary sub-loops: 1) a foraging loop, which emphasizes gathering relevant information, and 2) a synthesis loop, which emphasizes the forming of justified hypotheses. A particularly challenging cognitive task, *dual search*, occurs between these two loops in which analysts must create multiple alternative competing hypotheses based on the given data and, simultaneously, find additional relevant information that supports the created hypotheses.

Inspired by these and other sensemaking models, the visual analytics community has developed techniques and systems to support expert sensemaking. These include techniques to help users visualize and manipulate data at different levels of granularity, identify relationships between objects in large datasets, construct alternative hypotheses, and collect and arrange data and notes for later reference [9, 12, 28, 29] [2]. In contrast, we explore how novice crowds can support expert sensemaking. We focus on the sensemaking loop's "read and extract" step [26], which provides nuggets of evidence that can be used by experts to draw inference or trigger new hypotheses or search in later stages.

2.1.1 Collaboration among Experts. Collaboration among small groups of expert analysts is common practice in intelligence agencies. Beyond information processing, communication patterns and group dynamics introduce additional complexity [3, 12]. Researchers have sought to model collaborative sensemaking processes, including task decomposition granularity and team sizes from pairs of analysts [3, 5, 17, 31] to small teams [5, 11, 23]. Time zone and geographic differences also influence how analysts share information and pick up from each other's previous analysis. Goyal et al. [17] proposed and evaluated an interface for distributed sensemaking in real time, which improves task performance without increasing cognitive workload via implicit information sharing. Zhao et al. [36] developed a system that supports handoff in asynchronous collaborations through knowledge transfer graph, through an interactive tool with rich annotation features.

These theories of sensemaking and systems built to assist individual experts and small groups inspire us to bring much larger crowds into the sensemaking loop. However, unlike most work in this area, we recruit paid crowd workers who are non-experts and contribute for short time periods. We explore how to break down the dataset and delegate appropriate tasks to these novice transient crowd workers so that they can collaborate asynchronously and contribute meaningfully to the process of intelligence analysis.

2.2 Crowdsourced Text Analysis

Crowdsourcing, either alone or combined with automated approaches [16, 30, 34], has been used to forage and synthesize information with unknown topics and diverse structures [19, 20, 25]. In this section, we review prior research on crowdsourcing approach for text analysis.

One of the challenges of crowdsourcing is that it often lacks global context for local tasks. Researchers have developed iterative task designs to address this issue for text analysis tasks, such as clustering and categorizing [1]. Cascade [8] produces crowdsourced taxonomies of hierarchical data sets by letting workers generate, and later select, multiple categories per item. Frenzy [7] is a web-based collaborative conference session organizer that elicits conference paper metadata by letting crowd workers group papers into sessions using an asynchronous clustering tool. Alloy [6] leverages machine learning with clustering tasks in global contexts. In the Knowledge Accelerator [18], crowd workers are able to find relevant information about a given topic and aggregate the findings together meaningfully. Using context trees [32], crowd workers can provide ratings on global importance of documents via local views.

We draw inspiration from these projects, particularly the notion of integrating micro-tasks into a more collaborative, unstructured interface embodied in Frenzy and other forms of crowdware [35]. Unlike these projects, however, we focus on leveraging crowds to support analysts in finding hidden connections between entities in textual data, requiring contextual analysis of the contents and going beyond common sense knowledge.

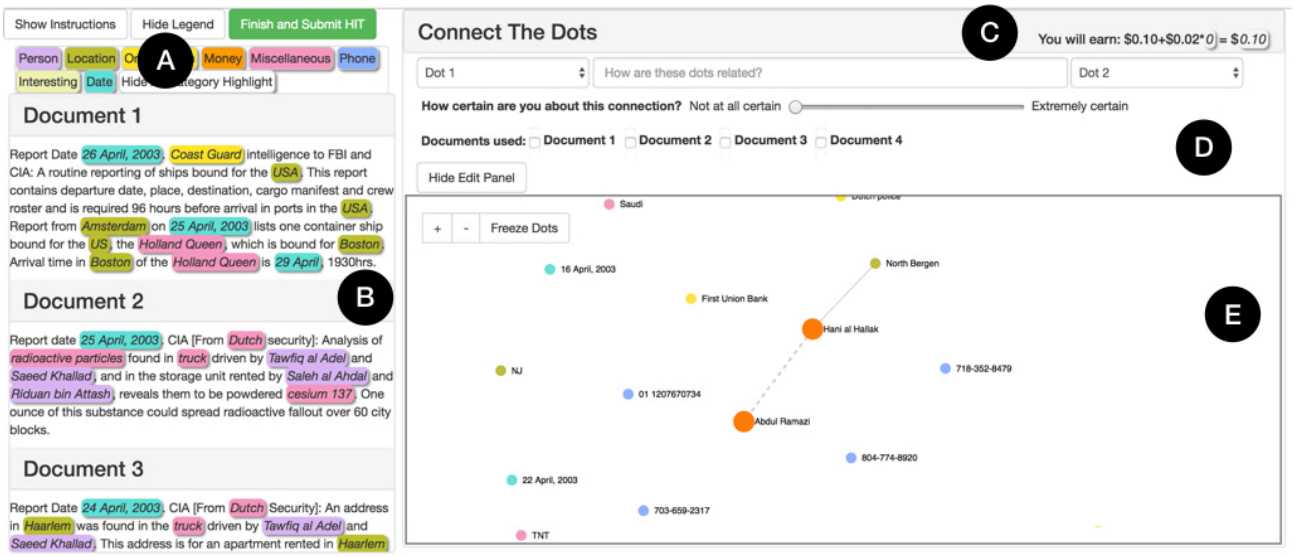

Figure 1. The Connect the Dots web application interface.

2.2.1 Named Entity Recognition and Crowdsourcing. Identifying and classifying the key entities (people, places, organizations, etc.) mentioned in text documents is a valuable early step to enable more complex information processing and sensemaking. Automated approaches to named entity recognition (NER) have made significant headway (e.g. [15][14][4, 10]), but human intervention is often required to achieve acceptable results in specialized contexts like intelligence analysis. Detecting semantically similar entities in textual descriptions can be complicated even for humans, and beyond the capability of machine-based approaches [22]. In the Linked Open Data (LOD) community, researchers have sought to bridge the gap between algorithmic matching and manual techniques by parsimoniously using human workers to guide the automated process of linking entities in natural language texts to existing structured concepts [13]. Other efforts seek to employ paid crowdsourcing as part of a human-in-the-loop workflow. For example, Wang et al. proposed hybrid human-machine approach called *CrowdER* [33], that uses machine-based techniques to make a first pass, and only ask crowds to verify more difficult pairs.

In this paper, we build on these earlier efforts and consider how crowdsourcing could support more complex entity recognition and identify more semantically distant entity relationships. We use existing techniques for a first pass at entity extraction, and then use context slices to allow crowds to find more subtle connections (e.g. the same terrorist suspect using two different aliases).

3 SYSTEM DESCRIPTION

To help crowd workers analyze documents and make connections between entities, we designed a web application called *Connect the Dots* (Figure 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.

3.1 Document View

The left side of the interface lists all the documents in the context slice (Figure 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 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.

3.2 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 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 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 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.

4 STUDY

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

- RQ1: What types of connections does the crowd create?
- RQ2: How do different slicing methods influence the crowd results?
- RQ3: When using context slices, how well can crowds find the connections needed for the solution?
- RQ4: When using context slices, how can we distinguish or prioritize the most important entities?

4.1 Dataset

We use a subset of the *Sign of the Crescent* dataset [15]developed for the purpose of training professional intelligence analysts. The original dataset consists of 41 fictional text intelligence reports regarding three coordinated terrorist plots in three US cities. Each plot involves a group of at least four suspicious people. Each report, or *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 [24]. 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 paper 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.

4.2 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.

4.3 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.

4.4 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.

4.5 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 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.

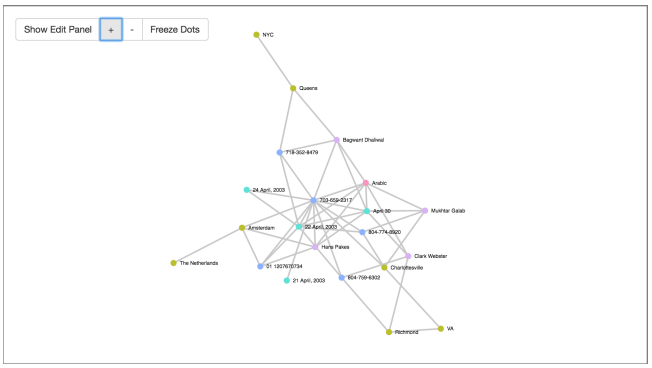

Figure 2. Example subgraph of connections made by five crowd workers for one context slice.

4.5.1 RQ1: 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 [15], we identified three types of connections:

T1. 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*.

T2. 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*.

T3. 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.

4.5.2 RQ2: 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).

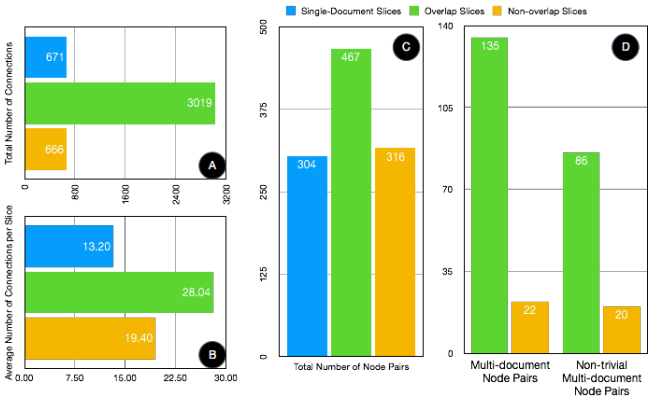

Figure 3. Average number of connections per slice, and multi-document node pairs in different slicing methods.

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 (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. A, C).

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.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. 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.

4.5.3 RQ3: 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 $O = C \cap G$. The precision value is then computed as $P = \frac{|O|}{|C|}$ and the recall value is $R = \frac{|O|}{|G|}$.

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 i^{th} context slice with threshold t be C_i^t ; the set of combined connections given a threshold t is $C^t = \bigcup_i C_i^t$. Precision, recall, and f-measure (harmonic mean of precision and recall) for each slicing method using combined connections in each threshold are shown in Figure 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 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.

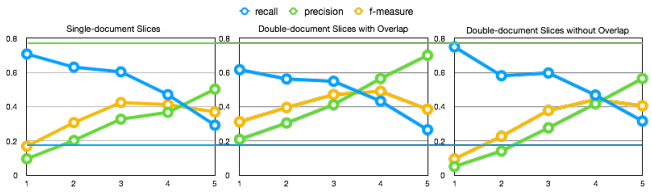

Figure 4. Precision, recall, and f-measure values for varying worker vote thresholds.

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 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 [9] 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 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.

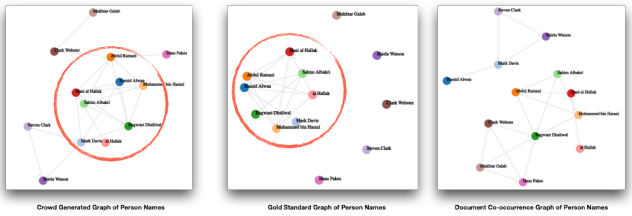

Figure 5. Relationship among person names by crowd workers, gold standard, and document co-occurrence baseline.

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).

4.5.4 *RQ4: 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 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 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.

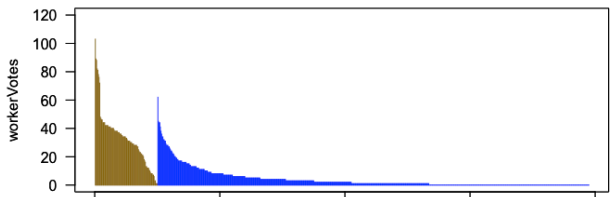

Figure 6. Ranked entity pairs by the number of workers connecting them. Gold lines are entity pairs from the gold standard. Blue lines are other possible entity pairs.

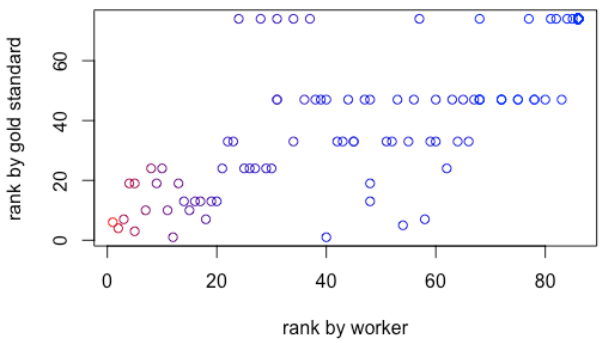

Figure 7. Rank entities by their degrees in the graph.

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 #1 and the entity with the fewest (if not 0) links is ranked #85. Figure 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.

5 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.

5.1 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.

5.2 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.

5.3 Sources of Strategic Information Retrieval

In Figure 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 relationships 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.

5.4 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.

5.5 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.

6 CONCLUSION

In this work, we explored non-expert crowds' potential to support a complex sensemaking process of expert analysts. Our results indicate that crowdsourcing offers a promising opportunity in collaborative sensemaking by bringing the crowds into the sensemaking loop. We found that with the concept of "context slice", 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 (3 or 4 votes out of all 5 crowd workers, depending on slicing methods), we can achieve good values of both precision and recall. Crowd-generated connections can be strategically retrieved and schematized by experts to provide deep insights. Last but not least, applying clustering algorithm to crowd-generated description labels can generate meaningful keywords that accurately describe the relationship between entities and help experts quickly understand the information without reading the original documents. This indicates the potential for providing condensed contexts and suggesting an effective starting point for more efficient investigation by experts.

ACKNOWLEDGEMENTS

Anonymized for review.

REFERENCES

[1] André, P., Kittur, A. and Dow, S.P. 2014. Crowd Synthesis: Extracting Categories and Clusters from Complex Data. *Proceedings of the 17th ACM Conference on Computer Supported Cooperative Work & Social Computing* (New York, NY, USA, 2014), 989–998.

[2] Andrews, C., Endert, A. and North, C. 2010. Space to Think: Large High-resolution Displays for Sensemaking. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2010), 55–64.

[3] Arias-Hernandez, R., Kaastra, L.T. and Fisher, B. 2011. Joint action theory and pair analytics: In-vivo studies of cognition and social interaction in collaborative visual analytics. *Proceedings of the 33rd Annual Conference of the Cognitive Science Society* (2011), 3244–3249.

[4] Bilenko, M. and Mooney, R.J. 2003. Adaptive Duplicate Detection Using Learnable String Similarity Measures. *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (New York, NY, USA, 2003), 39–48.

[5] Bradel, L., Endert, A., Koch, K., Andrews, C. and North, C. 2013. Large high resolution displays for co-located collaborative sensemaking: Display usage and territoriality. *International Journal of Human-Computer Studies*. 71, 11 (Nov. 2013), 1078–1088. DOI:https://doi.org/10.1016/j.ijhcs.2013.07.004.

[6] Chang, J.C., Kittur, A. and Hahn, N. 2016. Alloy: Clustering with Crowds and Computation. *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2016), 3180–3191.

[7] Chilton, L.B., Kim, J., André, P., Cordeiro, F., Landay, J.A., Weld, D.S., Dow, S.P., Miller, R.C. and Zhang, H. 2014. Frenzy: Collaborative Data Organization for Creating Conference Sessions. *Proceedings of the 32Nd Annual ACM Conference on Human Factors in Computing Systems* (New York, NY, USA, 2014), 1255–1264.

[8] Chilton, L.B., Little, G., Edge, D., Weld, D.S. and Landay, J.A. 2013. Cascade: Crowdsourcing Taxonomy Creation. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2013), 1999–2008.

[9] Chin, G., Jr., Kuchar, O.A. and Wolf, K.E. 2009. Exploring the Analytical Processes of Intelligence Analysts. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2009), 11–20.

[10] Christen, P. 2008. Febrl: A Freely Available Record Linkage System with a Graphical User Interface. *Proceedings of the Second Australasian Workshop on Health Data and Knowledge Management - Volume 80* (Darlinghurst, Australia, Australia, 2008), 17–25.

[11] Chung, H., Chu, S.L. and North, C. 2013. A Comparison of Two Display Models for Collaborative Sensemaking. *Proceedings of the 2Nd ACM International Symposium on Pervasive Displays* (New York, NY, USA, 2013), 37–42.

[12] Convertino, G., Billman, D., Pirolli, P., Massar, J.P. and Shrager, J. 2008. The CACHE Study: Group Effects in Computer-supported Collaborative Analysis. *Computer Supported Cooperative Work (CSCW)*. 17, 4 (Aug. 2008), 353–393. DOI:https://doi.org/10.1007/s10606-008-9080-9.

[13] Demartini, G., Difallah, D.E. and Cudré-Mauroux, P. 2012. ZenCrowd: Leveraging Probabilistic Reasoning and Crowdsourcing Techniques for Large-scale Entity Linking. *Proceedings of the 21st International Conference on World Wide Web* (New York, NY, USA, 2012), 469–478.

[14] Elmagarmid, A.K., Ipeirotis, P.G. and Verykios, V.S. 2007. Duplicate Record Detection: A Survey. *IEEE Transactions on Knowledge and Data Engineering*. 19, 1 (Jan. 2007), 1–16. DOI:https://doi.org/10.1109/TKDE.2007.250581.

[15] Finkel, J.R., Grenager, T. and Manning, C. 2005. Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling. *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics* (Stroudsburg, PA, USA, 2005), 363–370.

[16] Gomes, R., Welinder, P., Krause, A. and Perona, P. 2011. Crowdclustering. *NIPS 2011* (2011).

[17] Goyal, N. and Fussell, S.R. 2016. Effects of Sensemaking Translucence on Distributed Collaborative Analysis. *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing* (New York, NY, USA, 2016), 288–302.

[18] Hahn, N., Chang, J., Kim, J.E. and Kittur, A. 2016. The Knowledge Accelerator: Big Picture Thinking in Small Pieces. *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2016), 2258–2270.

[19] Heer, J. and Agrawala, M. 2008. Design Considerations for Collaborative Visual Analytics. *Information Visualization*. 7, 1 (Mar. 2008), 49–62. DOI:https://doi.org/10.1145/1391107.1391112.

[20] Kittur, A., Peters, A.M., Diriye, A. and Bove, M. 2014. Standing on the Schemas of Giants: Socially Augmented Information Foraging. *Proceedings of the 17th ACM Conference on Computer Supported Cooperative Work & Social Computing* (New York, NY, USA, 2014), 999–1010.

[21] Klein, G., Phillips, J.K., Rall, E.L. and Peluso, D.A. 2007. A data-frame theory of sensemaking. *Expertise out of context: Proceedings of the sixth international conference on naturalistic decision making* (2007), 113–155.

[22] Köpcke, H., Thor, A. and Rahm, E. 2010. Evaluation of Entity Resolution Approaches on Real-world Match Problems. *Proc. VLDB Endow.* 3, 1–2 (Sep. 2010), 484–493. DOI:https://doi.org/10.14778/1920841.1920904.

[23] Mahyar, N. and Tory, M. 2014. Supporting Communication and Coordination in Collaborative Sensemaking. *IEEE transactions on visualization and computer graphics*. 20, 12 (Dec. 2014), 1633–1642. DOI:https://doi.org/10.1109/TVCG.2014.2346573.

[24] On Cost-Effective Incentive Mechanisms in Microtask Crowdsourcing - IEEE Journals & Magazine: *http://ieeexplore.ieee.org/abstract/document/6704771/*. Accessed: 2017-10-01.

[25] Paul, S.A. and Morris, M.R. 2009. CoSense: Enhancing Sensemaking for Collaborative Web Search. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2009), 1771–1780.

[26] Pirolli, P. and Card, S. 2005. The sensemaking process and leverage points for analyst technology as identified through cognitive task analysis. *Proceedings of international conference on intelligence analysis* (2005), 2–4.

[27] Russell, D.M., Stefik, M.J., Pirolli, P. and Card, S.K. 1993. The Cost Structure of Sensemaking. *Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems* (New York, NY, USA, 1993), 269–276.

[28] Shrinivasan, Y.B. and van Wijk, J.J. 2008. Supporting the Analytical Reasoning Process in Information Visualization. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2008), 1237–1246.

[29] Stasko, J., Görg, C. and Liu, Z. 2008. Jigsaw: Supporting Investigative Analysis through Interactive Visualization. *Information Visualization*. 7, 2 (Jun. 2008), 118–132. DOI:https://doi.org/10.1057/palgrave.ivs.9500180.

[30] Tamuz, O., Liu, C., Belongie, S., Shamir, O. and Kalai, A.T. 2011. Adaptively Learning the Crowd Kernel. *arXiv:1105.1033 [cs]*. (May 2011).

[31] Tang, A., Tory, M., Po, B., Neumann, P. and Carpendale, S. 2006. Collaborative Coupling over Tabletop Displays. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2006), 1181–1190.

[32] Verroios, V. and Bernstein, M.S. 2014. Context Trees: Crowdsourcing Global Understanding from Local Views. *Second AAAI Conference on Human Computation and Crowdsourcing* (Sep. 2014).

[33] Wang, J., Kraska, T., Franklin, M.J. and Feng, J. 2012. CrowdER: Crowdsourcing Entity Resolution. *Proc. VLDB Endow.* 5, 11 (Jul. 2012), 1483–1494. DOI:https://doi.org/10.14778/2350229.2350263.

[34] Yi, J., Jin, R., Jain, A. and Jain, S. 2012. AAAI Workshop - Technical Report. (2012).

[35] Zhang, H., Law, E., Miller, R., Gajos, K., Parkes, D. and Horvitz, E. 2012. Human Computation Tasks with Global Constraints. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2012), 217–226.

[36] Zhao, J., Glueck, M., Isenberg, P., Chevalier, F. and Khan, A. 2017. Supporting Handoff in Asynchronous Collaborative Sensemaking Using Knowledge-Transfer Graphs. *IEEE Transactions on Visualization and Computer Graphics*. PP, 99 (2017), 1–1. DOI:https://doi.org/10.1109/TVCG.2017.2745279.