

# Bringing Collaboration to Intelligence Analysis: A Classroom Study

Leave Authors Anonymous  
for Submission  
City, Country  
e-mail address

Leave Authors Anonymous  
for Submission  
City, Country  
e-mail address

Leave Authors Anonymous  
for Submission  
City, Country  
e-mail address

## ABSTRACT

Intelligence analysis involves dynamic modeling and exploration of complex information spaces, and is fundamentally collaborative at any non-trivial scale. That said, the tools supporting intelligence analysis either do not support such dynamic workflows, or they do not support collaboration at all. Along these lines, we developed *CAanalytics* to address these two critical challenges by: enabling interleaving of data modeling and analysis within a single workspace; and supporting collaboration through shared annotation and increased activity awareness. This paper describes a classroom study of students training to become intelligence analysts and examines their tool usage over multiple usage sessions. We report the interleaving work patterns students spontaneously employ with our tool, as well as effect of the collaboration features on their teamwork. We also note the distinctions among different analytic and collaborative strategies. The paper concludes with design implications for intelligence analysis as well as other collaborative information analysis tools.

## Author Keywords

Intelligence analysis; collaboration; visualization; classroom study; activity awareness

## ACM Classification Keywords

H.5.3. Information Interfaces and Presentation (e.g. HCI): Computer-supported cooperative work

## INTRODUCTION

Intelligence analysis is a form of sensemaking wherein a team of analysts identify and evaluate causal relationships in a complex corpus of documents. A common example of such an analysis is the investigation of a crime, where a variety of putative facts are assembled, e.g. financial records, witness observations and interviews, and the social connections between persons of interest. Armed with these different assembled facts investigators collaboratively: assess means, motives, and opportunities; articulate and investigate further hypotheses and deductions; and develop one or more theories of the crime.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

CHI'16, May 07–12, 2016, San Jose, CA, USA

© 2016 Copyright held by the owner/author(s). Publication rights licensed to ACM. ISBN 123-4567-24-567/08/06...\$15.00

DOI: [http://dx.doi.org/10.475/123\\_4](http://dx.doi.org/10.475/123_4)

According to Pirolli and Card's model [28], intelligence analysis starts with modeling data from textual documents, followed by representing these data models in various information artifacts, and developing them into hypotheses. Systems that currently support intelligence analysis are aimed at a single phase and therefore only support part of the overall analysis workflow. This imposes a clear boundary between each of these phases on the analysts. For example, the Analysis of Competing Hypotheses (ACH) tool assumes that data has been modeled, and that relevant evidence can be adduced appropriately to various hypotheses, but provides no structured support for either. Other techniques, such as Information Extraction and Weighting (IEW), help structure evidence modeling, but do not extend utilization of evidence to hypothesis generation. The unintended boundary between phases has the consequence that data modeled in one software cannot be effectively utilized in hypothesis development in another system. And analysts have to handoff, often via replicating the data in the new system, information between software systems, making it difficult to revisit and revise the data model. We therefore are motivated to develop an integrated workspace in which analysts can model and analyze data in one place, and we utilize the system as an experimental instrument to investigate analytic behavior afforded by such integration.

Any intelligence analysis activity, at least at a non-trivial scale, is fundamentally collaborative [9]. The intelligence community puts great value on collaboration. A report from the Director of National Intelligence, *Vision 2015*, called for “a dramatic shift from traditional emphasis on self-reliance toward more collaborative operations” [1, p.13]. However, most analytic tools that are widely used in the intelligence community (e.g. Analyst's Notebook [20] and PARC ACH [27]) do not support collaboration. The community has to rely on separate collaboration tools (e.g. email, Intellipedia [35]), which lack serious support for analytics, for team coordination [34]. Thus, analysts must coordinate their work outside of their tool support, manually sharing their analytic products.

Supporting collaboration in intelligence analysis is challenging, and perhaps more than other situations (e.g. collaborative writing, wiki) because the task itself can be extremely complex. A team could be working with much more complex data structure (e.g. spatial data, temporal data, and relational data, as opposed to text only), coordinating through multiple analytic artifacts (e.g. map, timeline, network, as opposed to only document only), and making sense of different levels of analysis, assumptions, and hypotheses, both synchronously and

asynchronously throughout a long-term course of collaborative interaction. This study aims to investigate how technology can mediate team collaboration in a sufficiently complex analytic task over multiple usage sessions.

We performed our study within the context of an intelligence analysis course. This classroom study provides a natural environment in which participants engage in multi-session, relatively complex class projects. Due to the difficulty in accessing professional analysts, or having them in longitudinal design loops, *analysts-in-training* are a good compromise. These students already have some knowledge and experience with state-of-the-art analytic techniques and tools. In class projects, students are graded on their ability to understand and enact the professional practices of information analysis. This strong normative emphasis on problem solving practices provides us an appropriate evaluation context for new interactive tools: Tools are only valuable to the students insofar as they actually support better practices and better outcomes.

We are thus motivated to investigate the feasibility, effectiveness and consequence of supporting collaborative intelligence analysis in the context of classroom study. We have developed a tool that includes annotation for data modeling, interactive visualization for data analysis, and a set of collaboration features. Our contributions include 1) evaluating tool support, in particular, support for data modeling and analysis and support for collaboration, in a classroom environment; 2) identifying collaboration characteristics and issues in team process; and 3) suggesting design implications for future collaborative intelligence analysis support.

## RELATED WORK

### Intelligence analysis workflow

Pirolli and Card's Think Loop Model [28] is a widely used model of intelligence analysis. The model describes the process of information foraging and sensemaking in which raw evidence is successively modeled, filtered, and synthesized into a best hypothesis. The model is a bottom-up process of structure building, but also includes a local feedback loop at each stage. Thus, analysts can reconsider propositions in the evidence file, asking how they are related, or a given hypothesis, asking what schemata it rests upon.

Empirical studies of information analysis suggest that the iterative looping can have wider scope than is obvious in the Pirolli and Card's model [28]. For example, Chin et al. [8] observed five professional intelligence analysts working both individually and as a team. They found that analysts often need to review the original documents even at advanced stages of analysis. The scope of these reconsiderations is not consistent with the local feedback architecture of the Think Loop Model. It seems more consistent with a parallel or multi-phased model [38] in which structure building occurs at a variety of levels in parallel. Similar findings were reported by other empirical studies [21, 22, 18].

However, tools supporting intelligence analysis are often designed targeted at a single phase of activity, and thus not supporting the whole workflow. Many analytic tools assume

data model has been finalized and ready to be visually analyzed, and affords no utility to construct or refine data models. As Ware termed as "*asymmetry in data rates*" [36, p.382], analytic tools emphasized data flowing from systems to users far more than from users to systems. Functionalities are mostly designed to adjust data representation rather than modeling, which are in fact equally important in intelligence analysis. A report from a workshop of professional intelligence analysts listed "dynamic data processing and visualization" as one of top requirements in computational support [2], and Kang et al. [22] made another call from findings of their empirical study.

### Collaboration and awareness in intelligence analysis

Collaboration is critical in intelligence analysis. The intelligence community has called for collaboration, and indeed closer collaboration, as opposed to merely coordinating draft products by the end [1]. Empirical studies [8, 22] reported that analysts' practice was fundamentally collaborative.

However, computational tools supporting intelligence analysis either do not do support collaboration, or they do not integrate serious analytic support. On one hand, not most analytic tools (e.g. Analyst's Notebook and PARC ACH) are designed for single user only, with only a few exceptions (e.g. Te@mACH [11]). On the other hand, collaboration tools only support general coordination between teams and agencies, without analytic functionalities, as Treverton [34] reviewed collaboration tools currently used in the intelligence community. For example, Intellipedia [35] is a wiki platform for sharing of intelligence reports and documents, yet it does not support data modeling and analysis at all.

A key enabler for effective collaboration is *activity awareness*, defined as team's awareness of its own sustained collaborative activity [6, 7]. Derived from Activity Theory [23], activity awareness encompasses information covering all aspects of collaboration, such as partner presence, mediating artifacts, group actions, social interactions, shared information, and group values and norms. Activity awareness has been utilized as a design concept in guiding and evaluating collaboration features.

Many lab studies have been reported to investigate specific awareness features to support collaboration in intelligence analysis. For example, Convertino et al. [9] examined the use of public and private views for role based collaboration. Goyal and Fussell [13] studied the effect of hypotheses sharing on sensemaking. Mahyar and Tory [24] designed a visualization to connect collaborators' common findings and evaluated its support for team performance. Hajizadeh et al. [15] explored how sharing teammate's interactions affects awareness. These studies provide evidence to validate hypotheses of specific design features. However, due to time constraint (mostly within one hour), these studies had to employ a simplified task with reduced content and complexity. Artifacts created by participants were thus relatively simple and superficial (e.g. with a single artifact or few items in an artifact). More complex task would have pushed participants to create more sophisticated artifacts (e.g. multiple views or cluttered display that requires filtering) and to try balancing between team coordination and individual analysis, which would have provided more insights

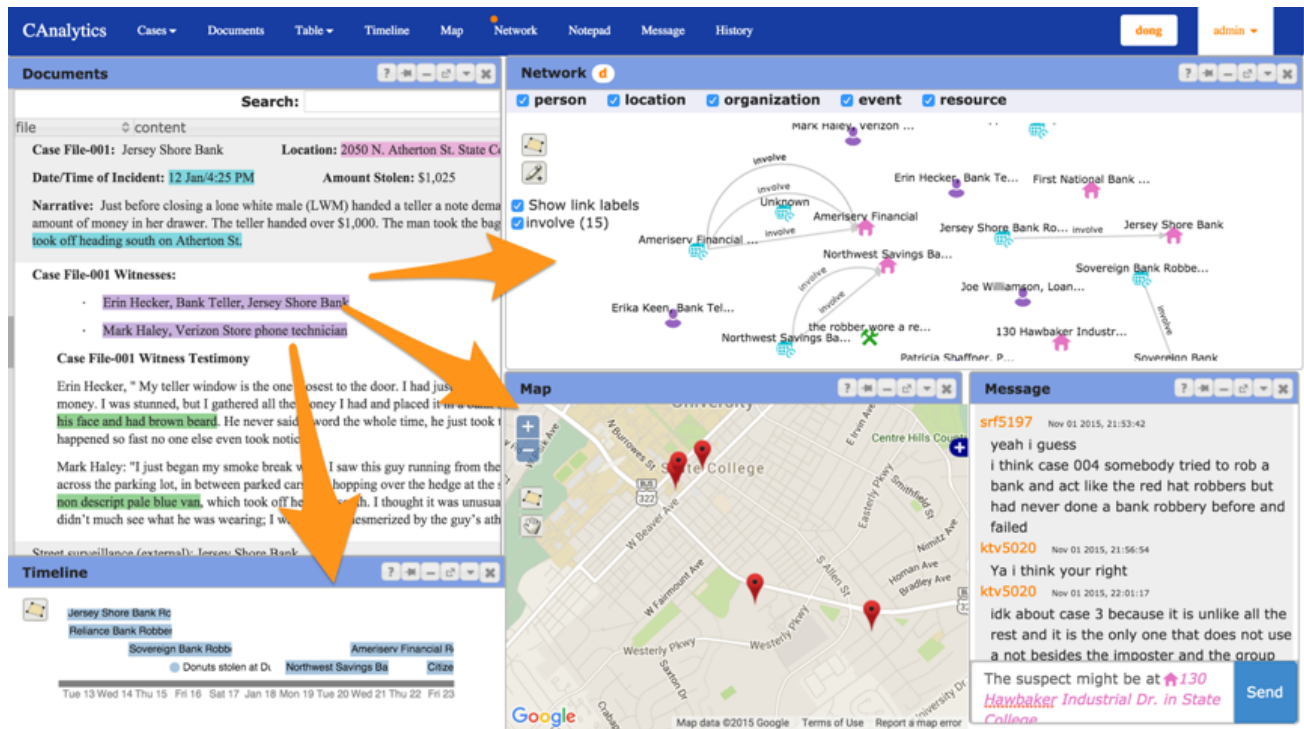


Figure 1. CANALYTICS user interface. Each window is closable, movable, and resizable. Shown here are *document* view (top left), *timeline* (bottom left), *network* (top right), *map* (bottom middle), and *message* (bottom right). Other windows include *table*, *notepad*, and *history* view.

into team-based analytic process. Our study aims to examine tool usage in a course project, which allows for higher task complexity and longer teamwork time, thus gaining more insight into the team process.

## CANALYTICS FEATURES

As shown in Figure 1, we developed a web application tool, *CANALYTICS* (standing for “C-ollaborative Analytics”), to support teams of analysts in identifying, visualizing, integrating and assessing facts from multiple sources. Two particular design objectives are 1) to build an integrated environment for intelligence analysis, and 2) to support collaboration with shared data and awareness functions. The design is informed by earlier paper prototype studies [4, 5], in which the researchers examined team’s communication patterns and spontaneously created artifacts in a crime scenario. We also take into account findings from empirical studies conducted by Chin et al. [8] and Kang and Stasko [22] when making design decisions.

### *Coordinated multiple views for data modeling and analysis*

To enable an integrated environment for different functionalities of analysis, *CANALYTICS* employs a multiple-view interface, with each view in a floating, closable window. Views are coordinated and share the underlying data pool so that products of different activities are not fragmented. The following paragraphs describe features of each view.

The *document* view supports evidence modeling through annotation. In the document view users can select and highlight a snippet of text and annotate it as a type of entity such as a person, location, event, etc., or as a relationship between entities. Unlike other entity-based systems such as [3, 33], we

use annotations to allow analysts to manually create evidence objects of interest. Manual annotation allows for greater user control in terms of information of interest and granularity that best suits their ad-hoc analytic needs. Users can add attributes to the annotated object, e.g. time in an event, and coordinate in a location. Users can also make reference to other objects in the attribute; for example, users can add people objects to an event indicating that these people were involved in the event.

The data modeled through annotation is then displayed in visualizations in the same workspace, including *table*, *timeline*, *map*, and *network* view—artifacts that are frequently constructed to hold attribute data, temporal data, spatial data and relational data respectively [5]. Figure 1 shows an example: when an annotation is created in the document view with information about time, location, participants, and their relationships, a new event is created in the timeline view, a new location is created in the map view, and new people are added to the network graph with a labeled edge representing the relationship (or new edges are added to existing nodes). Hovering the mouse over an entity will activate an entity detail window that displays attributes in detail, and analysts can modify, or re-model the entity in situ.

The views afford brushing and linking interactions; that is, when users brush entities in one view, related entities are displayed in other views. Thus the analyst can narrow down entities to their interest by: specifying a time range using the functionality provided on the timeline; making a spatial query with map filter; or selecting a cluster of entities by drawing a bounding area in the network view.

### Collaboration and awareness features

To support collaboration, CANalytics affords real-time collaborative editing, similar to the Google Tools. Users can open several concurrent editors to collaboratively edit multiple entity objects. Entities and annotations are immediately shared within a team and rendered in teammates' corresponding views.

In addition to real-time data sharing, CANalytics supports other awareness features. A *notification system* sends individual's actions to the team in the form of a text box located in the top right corner of the workspace. An iconic indicator on top of a view window, which we call *tool coordinator*, shows who else is also working on this view. A *message tool* is a real time chat window that enables team communication with persistent message history. The system also maintains a traceable log of time-stamped individual activities in the *history tool*. Users can learn team activities about who did what to which object at what time. Entities mentioned in the message tool and history tool are hyperlinked and will trigger pop-up detail window when the user moves the mouse over them. With these awareness features, users who work synchronously can be informed of others' activity continuously; users who work asynchronously will be able to use the history to reconstruct their work status and become aware of changes beyond the point of their last interaction.

We also included a *notepad*, a collaborative text editing tool, to support hypothesis development. We integrated Etherpad<sup>1</sup>, an open-source collaborative editor similar to Google Doc, for teams to compose their hypotheses. Users can insert tables (e.g. an ACH matrix) and images (e.g. screenshots of the tool views).

### SETTING AND METHOD

The setting for this study was an undergraduate course in an intelligence training program in a US university. The program was designed to train students to become professional intelligence analysts. A key requirement of the course is to emphasize hands-on practice on team-based intelligence analysis.

During the first ten weeks of the course students learned several analytic techniques, including IEW (a technique to extract evidence and assess their values), ACH (a technique to evaluate multiple hypotheses against evidence), timeline analysis and network analysis, as well as state-of-the-art tools to implement these techniques, including Analyst's Notebook and PARC ACH. Students also practiced applying these techniques in two hands-on projects before our study. A typical workflow started with IEW to extract and model evidence from documents, followed by building analytic artifacts such as an ACH Matrix in PARC ACH, and a timeline and a network graph in Analyst's Notebook. Since data from IEW table could not be shared or extended to other tools, students had to replicate data for each different tool. Most tools they used lacked serious collaboration support (except that some teams used Google Doc to construct an IEW table). Analysts were unable to contribute simultaneously (an issue known as production blocking

<sup>1</sup><http://etherpad.org>



Figure 2. Classroom setting

[10]). Students often divided their work by tools: each person picked a tool and then created and analyzed an artifact with the tool on their own. This had the consequence that findings and hypotheses be made without integrating collective efforts and diverse knowledge. Analysts coordinated work by manually sharing documents or graphs through email or cloud storage service (e.g. Dropbox), resulting in a scattered placement of results, requiring repeated manual resynchronizing to identify redundant or missing pieces of information, analysis of information, and analytic hypotheses. In summary, the students in our study had learned and practiced with tools lack of support for collaboration, and were all aware of the shortcomings.

Of the 98 students enrolled in the course (from two sections), 73 consented to participate in the study. All of the 73 students held major in the program of Security and Risk Analysis. Most (75%) of them were in the third academic year ( $M = 3.06$  years,  $SD = .44$ ), indicating that participants in our study had relatively advanced experience and knowledge in intelligence analysis. Participants' age ranged from 19 to 28 ( $M = 20.4$ ,  $SD = 1.18$ ). 77% of the participants were male.

Students were given a tutorial on CANalytics a week before the project began. One of the authors walked through the features of CANalytics, without enforcing or implying any strategic use of the tool. Students then accomplished a small case analysis at their own pace. During the week of the actual study, one author was constantly available to help with any technical issues. Although students were encouraged to make full use of CANalytics, to ensure a naturalist environment students were always free to employ any other tools that they believed useful.

Our study began on the 10th week of the course and lasted for one week. The analysis that students performed on our tool was the investigation of a series of bank robberies, all of which was fabricated by the course instructor. Teams were provided a set of documents pertaining to seven robberies, including police reports, witnesses reports, video records, and news media. The analysis was designed to be open-ended, meaning that there was no single, definitive answer. The instructor explained that the task was to simulate real world scenarios, in which analysts always reasoned in the circumstances of uncertainty, ambiguity, and complexity. The instructor told



the students that he expected the analysis to take around 6 hours, including in-class and outside-class work. In the end of the project students were required to submit a team report, describing their hypotheses, assumptions, conclusions and supporting evidence. When in class, students used a 27-in Macintosh desktop (as shown in Figure 2. Students could use any equipment when outside class.

Students were randomly assigned into 25 teams (23 three-person teams and 2 two-person teams). Research suggested that group size be an important factor in group collaboration, thus two-person teams might behave differently from three-person teams. We thus excluded data from the two-person teams in our analysis in this paper. Also, from the log (and confirmed by their questionnaire), we found that one team made little use of CAnalytics and opted for other tools (Google Doc). Hence their data was also excluded. Thus in this paper we reported the result from 22 teams.

We employed several data collection approaches. We administered a post-study questionnaire. The questions used 7-point likert scale and included 9 items measuring individual's self-reported awareness, 7 items for communication quality, 6 items for collective efficacy, and 3 items for perceived performance ([9]). We also used NASA-TLX [16] to measure cognitive load. The questionnaire also included open-ended questions asking how the tool helped or impeded their work. We captured user interactions with system logs. Instead of simply logging low-level events like mouse click and keyboard strokes, we recorded actions such as creating an annotation and deleting an entity. Finally, we reviewed team reports and graded them as an indicator of team performance. Since the task was open-ended, there was no single right answer. We constructed an assessment rubric together with the course instructor by listing all possible hypotheses and evidence from the documents, with a full score of 16. The first author and a research assistant graded the reports independently. If the grades differ by less than 2, an average is set as the final grade (14 out of 22 reports). Otherwise (the rest 8 reports), the two graders review the reports together and make an agreement.

## RESULTS

Over the week, teams created 1805 entities and 1529 relationships in total. The number of entities teams created ranged from 24 to 223 ( $M = 82, SD = 39.9$ ), and the number of relationships ranged from 7 to 237 ( $M = 69.5, SD = 51.0$ ), showing a large variety. The big range was related to different team data modeling strategy, which will be discussed in detail later. While teams could work on the project any time, we found three intensive usage sessions from the interaction log: two were in class and one was outside class, before the team report deadline.

CAnalytics was generally well received by the students. An overview of the related survey items (shown in Figure 3) shows that students positively rated all aspects of the tool except cognitive load, towards which they had a close to neutral feeling.

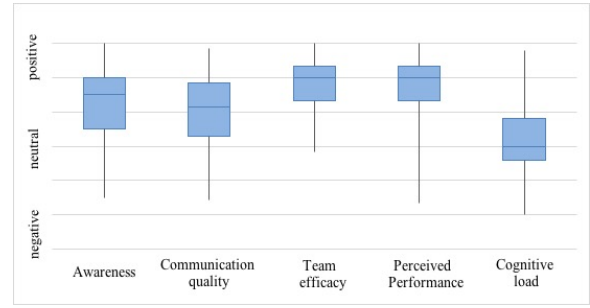


Figure 3. Survey responses (box shows Q1-Q3 and median; ends of whiskers show maximum and minimum)

## Interleaving data modeling and analysis

We examined the pattern of data modeling and analysis first by qualitatively looking at a visualization of the entire interaction log (e.g. Figure 4a shows one team's interaction). All teams worked intensively on data modeling as they started off the project. This was the phase when teams were getting themselves familiar with the documents and made initial annotation input into CAnalytics. Starting from certain time point, all teams started analysis on visualizations, followed by frequent transitions between data modeling and analysis. 11 teams started data analysis in the first usage session, while the other 11 teams had this transition in the second usage session. In average, the transitions occurred in 47.6 minutes after the project began. The earliest transition occurred in 14 minutes after the team started the project, and the last team had the transition around 104 minutes, later in the second session. We also found performance difference among teams that started analysis early and those late. Teams that started analysis in session one had higher performance ( $M = 8.6$ ) than teams that started from session two ( $M = 6.7$ ), although the difference was not statistically significant.

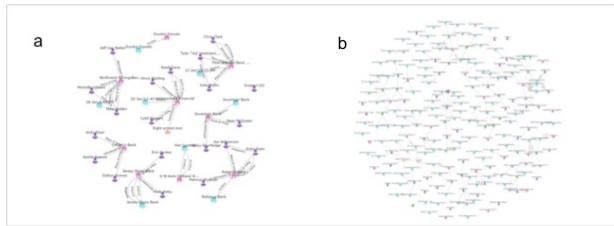
The fact that participants returned to making annotations after analysis indicated that they did not wait to start analysis till they had finished modeling. Indeed, the activity of data modeling and data analysis were highly interleaved throughout the project (as shown by the interleaving color bar in Figure 4a). Participants switched from one activity to the other activity frequently. The state transition diagram (Figure 4b) demonstrates the interleaving in an aggregated way, in which we encode the number of transitions as width of the link. This result confirms our design expectation that data modeling and analysis should not be supported as separate staged activities, and that an integrated environment should streamline the workflow.

## Data modeling: filtering vs. accretion

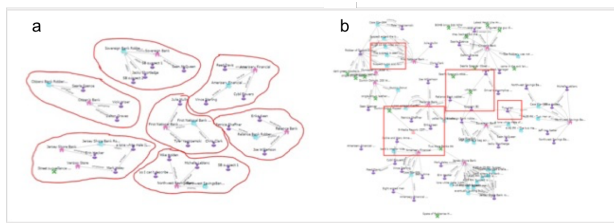
We noted a distinction between filtering and accretion strategies in data modeling, similar to what was reported in the paper prototype study [5]. Filtering is selectively modeling of data and adding to an artifact. Users must decide what information is relevant, and thus what is to be excluded, as well as what granularity of information is to model. Filtering requires more team coordination, because teammates must reach a common ground of the current problem as well as information needed to answer the problem. Figure 5a is an example of network built



**Figure 4.** (a) Visualization of interaction logs of Team 107. Each row of colored marks indicates the sequence of top level activities a participant performed. (b) State transition diagram of interaction logs of Team 107. Each node is an activity, whose size represents the time spent on the it (M: data modeling; A: data analysis; H: hypothesis development; C: coordination); a link represents a switch from one activity to another, whose width encodes the number of switches. We see highly frequent transitions between data modeling and data analysis



**Figure 5.** Network artifact comparison: filtering (a) vs. accretion (b)



**Figure 6.** Network artifact comparison: separate clusters (a) vs. connected clusters (b). The parts highlighted in red squares in (b) are key evidence that connects clusters

with filtering strategy. It only represented key information of robberies.

Accretion is an attempt to comprehensively represent the problem by adding all information to an artifact. Users extract every fact from the document, regardless of its immediate relevance to the problem. Accretion requires less coordination as it is relatively mechanical note taking. A disadvantage of accretion is that it can be time consuming to model all details and the produced artifact could be fairly complex. An example is Team 108, who modeled every step the suspects took, which resulted in far more entities (223) than the average (82) and more cluttered network view (Figure 5b). This accounted for the large range of entities created. Users also realized the problem. They reflected that they spent too much time in detail events, and many did not help their analysis at all:

*I felt that after we were done annotating, we hadn't really accomplished anything and that we were no closer to solving the case than when we had started. In the end it didn't really help that we had annotated the data. (P86)*

#### Artifact analysis: fact vs. inference

We examined the analytic artifacts teams created, the network graph in particular because social relationships played the most critical role in this specific scenario and teams spent most time on network analysis (as reflected from the log). We found a clear distinction among the network artifacts. For example,

networks from 8 teams (mean performance score = 7.8) consist of separate clusters (Figure 6a). Nodes within a cluster are connected, representing information space of a robbery case; Nodes between clusters are nonetheless not connected, indicating each robbery is a self-contained case. However, these teams did not miss the connections between robberies in their report. It turned out that these teams documented any possible relationships between robberies in the notepad tool as a list, separate from the network graph; that is, these teams distinguished information content and synthesized them in different artifacts.

In contrast, 6 other teams (mean performance score = 8.3) created networks composed of connected clusters. While a cluster is still a representation of a robbery, some of them are connected through an evidence node. An example is Figure 6b, in which we mark in red four *connectors* that link the clusters. These connectors were key evidence that led the teams to hypothesize that those robberies were related and might be committed by the same criminal group (e.g. the white van shown up in two robberies). These teams represented all information in one artifact.

By comparing these two types of networks, we found that links within a cluster were typically *factual* relationships modeled from raw documents (e.g. a white van was witnessed at a location), and links between clusters were often *inferences* beyond literally documented (e.g. a white van at location A is the same van witnessed at location B). Teams creating separate clusters represented only facts in the network and held evidence of uncertainty in a separate artifact. One advantage of distinguishing facts and inferences is that teams can always be aware of assumptions made when making a conclusion. And since all inferences are held in one place, teams are forced to confront them and review their uncertainty iteratively in the process. However, the strategy also adds difficulty to analysis as analysts may overlook or fail to combine evidence scattered in different artifacts.

On the contrary, in connected-cluster networks, facts and inferences overlaid in one artifact together drive the layout of the network, are better synthesized, and give analysts a clearer picture in one place. Teams may discuss and evaluate the level of uncertainty of inferences to decide whether to add them to the network. This strategy makes analysis more interactive among teammates: they need to negotiate, evaluate, and reach consensus on the value and validity of inferences. However, a problem with mixing facts and inferences is, to some extent teams might forget whether a link is factual or inferred, and

Element	Example
Social awareness <i>who is present?</i>	CAnalytics helped me stay aware,of my teammates activities because I could see who was logged on in the top,right corner (P123)
History awareness <i>Who has done what?</i>	The way you are able to view when and where your teammate made or updated annotations/information was the key to staying aware of what your team has done. It is a great tool in respects to that. For example, I was able to view the changes my team made while I was not using the CAnalytics tool at the same time they were using the history tab. (P171)
Information awareness <i>What is being changed?</i>	CAnalytics was very helpful in keeping us updated on what was being changed/noted/amended by whom and when. This was very beneficial for staying on the same page and knowing what changes were being made so no one individual was out of the loop. (P157)
Action awareness <i>Who is doing what?</i>	I liked how you could always see what your teammate were viewing on the website. For example I was working on the bluf when my teammates were working on the network part of the program. If I were to come across a piece of information that I thought might be helpful to them I would just tell them. My teammates did the same thing in return. (P51)
Intention awareness <i>Who is going to do what?</i>	CAnalytics showed what tab [tool] my teammates were working on which helped me be aware of what they were working on. For example, if I saw that one of my teammates was on the network tab, I knew that they were attempting to connect the information that was relevant to one another..I would then be able to mention any new findings I had that could influence their work (P160)

**Table 1. Subject feedback of awareness instances**

ask whether conclusion derived from the visualization can be trusted under uncertainty.

### Collaboration and awareness

One recurring theme in the subject feedback we collected was that the collaboration features were helpful for solving the problem. In the survey 88% of the students positively rated their group awareness. Participants appreciated that the tool complemented traditional analytic tools, describing CAnalytics as Analyst’s Notebook with real time synchronization features similarly to Google Docs, or described it as Google Doc with added visual analytics. To quote one participant, “*CAnalytics is like an analysts notebook that multiple people could work on at once [...and] an analysts version of a Google Doc*” (P65).

Participants reflected that they could now contribute simultaneously without concerns of interference and could have everything in one place instead of manually sharing documents via a cloud service.

*It was much easier to coordinate as a team with CAnalytics because we could all work on the same system at the same time. Without CAnalytics, we were forced to do the work separately and compile all the work onto one system after we had finished. (P156)*

Students also reported that being able to see teammate’s status made the task more motivating and engaging:

*During class I wasn’t sure if my teammates were doing work for that class or another thing but then seeing their dot [tool indicator] switch between applications on the software and updates pop up on my screen I knew they were doing work for 231. (P141)*

*The fact that you can see what other teammates are doing and they can see what you are doing creates a sense of accountability in terms of separating the work load. (P51)*

The motivating effect of awareness might account for, at least partially, the fact that teams were participating equally. We measured the equality of participation in terms of number of

created entities and time spent on CAnalytics. We refer to Olson [26] in calculating equality: one minus the variance of proportions times 100 (for better readability). Thus the score ranges from .67 (when only one person contributes in a three-member team) to 1.00 (when everyone participated exactly equally), and higher score indicates higher balanced participation. The resulted equality of created entities and time was .96 and .99 in average respectively. This indicates participants contributed fairly evenly.

Another repeated theme was the awareness features helped assure all teammates were executing the team plan. Participants reflected on their experience that a common team breakdown was misunderstanding of the team plan, and that they did not realize the misunderstanding until everyone had spent significant efforts finishing their “perceived” job. CAnalytics made plan execution assured because they could always see where teammates were working and what they were generating; and if anything unexpected happened, they could communicate immediately rather than in the end of the project.

Participants reported many other instances of awareness they realized using CAnalytics. We categorized them based on the element of awareness, or the essential problem of awareness of *what* [30], into social awareness, information awareness, action awareness, history awareness, and intention awareness, as shown in Table 1.

When asked what features helped them stay aware of team activities, 28 participants mentioned the tool coordinator, 24 mentioned the notification system, 19 mentioned the history tool, 14 mentioned the real-time update of user-generated data, 12 mentioned the collaborative editor, and 7 mentioned the message tool. Although the number of mentions does not simply indicate tool usefulness, it suggests users were explicitly aware of these awareness features and appreciated their support.

Students’ positive feedback on awareness was further corroborated by interaction logs. For example, we measured the number of entities accessed by other collaborators. While data generated by users is automatically shared, it is up to collaborators whether to read/edit the shared information or

ignore information altogether. The log showed that on average, 77.6% of the created entities were *read* by at least one other teammate. Further, We measured how many entities were *edited* by collaborators, a phenomenon we argue requires higher awareness, because the collaborator must not only realize the creation of the entity, but also understand its content. We defined *peer editing*, manipulated as the ratio of editing other’s entities over editing those created by oneself. We found that all teams edited collaborator’s entity objects, with a peer editing value equal to .83 ( $SD = .45$ ). The result suggests that teams had little difficulty accessing and modifying partner’s created data objects.

One major critique is the lack of support for sharing intermediate analytic insights. An insight is revealed and contextualized by a specific arrangement of views, e.g. a reduced data view of interest through filter, a highlighted entity representing the analytic focus, and a clustering layout of network to demonstrate a specific relationship. While teammates share the same data pool, they are likely to have different views of data, and thus different *interpretations* toward the data. A dynamic view together with its interpretation represents user’s intermediate analytic status. Sharing these insights could inspire team analysis [12]. With CAnalytics participants complaint that they could not easily make such communications. The team could “be looking at the same information but arranged in completely different ways” (P131).

### Labor division strategies

We noted different labor division strategies from interaction logs. Seven teams followed *document-based collaboration* (DBC): they divided their work by evenly distributing the different documents among team members (as shown in Figure 7a). Each member read through and made annotations on their own set of documents. An advantage of this strategy is that individuals get less workload and thus have more time to think deeply about their own documents. Individuals also implicitly take over the responsibility of their assigned documents to gain insights and share them when the team synthesize findings in later analysis. When the team needs information from one document, they rely on the “document owner” to share his/her finding. The failing of this strategy is thus in case an individual fails to identify or convey evidence in the document, the team may overlook the information altogether [4].

Four teams followed an *entity-based collaboration* (EBC) strategy. Instead of dividing by documents, they divided work by entity types: each individual went through all documents but only annotated entities of certain types, e.g. teammate A only annotated persons and teammate B annotated locations (as shown in Figure 7b). This strategy seems to save teammate’s time in the data modeling phase. And since each person focuses only on certain entities, they are more likely to identify recurring patterns, for example, the white van used in multiple robberies. However, focusing on certain entities could lead individuals to superficial syntactic scanning of documents instead of deep reading. This could further lead to extremes of annotating all entities of the type, whether they are related to the problem at hand or not, similar issues with accretion

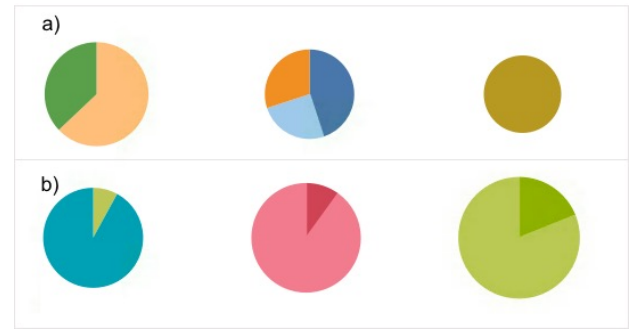


Figure 7. Pie charts showing different labor division strategies. Each pie chart corresponds to one team member. (a) Document-based collaboration. The pie chart shows the documents one team member annotated, color coded by document ID. (b) Entity-based collaboration. The pie chart shows the entity types one team member created, color coded by entity types.

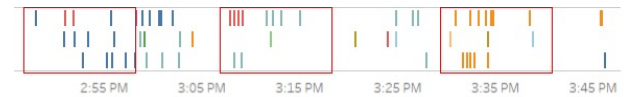


Figure 8. Graph showing the timeline of one team creating annotations. Each row corresponds to one team member. Each bar represents an annotation, color coded by document ID. The red blocks highlights the periods when all teammates worked on the same documents simultaneously.

strategy discussed beforehand. Indeed we found from the interaction log that EBC teams created more entities ( $M = 101$ ) in average than DBC teams ( $M = 94.5$ ). Moreover, with emphasis on certain entities, individuals are likely to know only partial aspects of a robbery and hence have difficulty connecting and synthesizing facts to deduce any conclusion. As one participant reflected, “we broke up by entity type, which reduced our individual involvement in each other’s entity types” (P99). The result indicated that the average performance of EBC teams ( $M = 7.63$ ) was lower than that of DBC teams ( $M = 9.25$ ), although the difference was not statistically significant.

The rest (eleven) of the teams did not show specific labor division patterns. Indeed, teams did not necessarily have to divide their work in order to collaborate, especially when the collaborative tool provides possibility to work closely together. Teammates could read and annotate the same document because they could see new annotations by others in real time and build on other’s annotation. Figure 8 shows an example where one team worked on the same document simultaneously. The three team members exhibited high synchronicity in which document to analyze, which we believe was not by accident. This has the advantage that teammates are always on the same page and can discuss hypotheses throughout the analysis process. Participants did have concern for possible duplication. As one participant complaint, “we could not actively see the changes our teammates were making until well after they had made them” (P46). This was because an annotation was shared only *after* it was created, yet another teammate might already be drafting an annotation on the same text snippet in the meantime.



### Interaction with team performance

To systematically evaluate factors that influenced team performance, we conducted a multiple linear regression between team performance as the dependent variable and team collaboration characteristics (i.e. equality of entities and time respectively, peer editing, number of entities created, shared entities, switch time from data modeling to analysis) as independent variables. We treated a team as a system [17] and modeled team-level relationships. The relationship of individual level variables (e.g. survey items) could not be simulated with regression directly because data within a team was inter-dependent. The analysis was performed using *R* [29].

The result is shown in Figure 9. The regression model was found to be statistically significant ( $F(5, 16) = 4.58; p < .01$ ), with an *R* squared equal to .59, meaning 59% of the variability in team performance was explained by the model. Five team-level variables contributed significantly to prediction of team performance. The model suggested that balanced contribution of entities predicted higher team performance scores ( $\beta = 45.79, p < .05$ ), but balanced participation time did not show such effect. More peer editing led to better performance ( $\beta = 3.89, p < .01$ ), implying that analysts should be encouraged to model data as a team, and to remodel other's entity objects as needed. Longer elapsed time before a team started analysis (*switch time* in the figure) predicted lower performance ( $\beta = -.05, p < .05$ ), which suggested that teams should shorten the time of pure data modeling and start analysis earlier. Larger number of entities teams created also predicted higher performance ( $\beta = .03, p < .05$ ). This can be interpreted that teams providing sufficient model input did benefit from system support. However, we are not ready to claim more entities are always better because we should also be cautious that entities irrelevant to the team problem bear no value to team analysis, as discussed beforehand. Somewhat surprisingly, the number of shared entities negatively predicted performance ( $\beta = -8.99, p < .05$ ). We considered this might also relate to the issue of entity value: sharing entities without relevance to the team problem leads to reduced team efficiency and creates distraction, which in turn would lead to worse team performance. Another possibility is that teammates may not necessarily access all entities partners created. The theory of Transactive Memory [37] suggests that teammates do not have to know everything in the team but should know "who knows what". Individuals specialize in their own field (e.g. one specializes in event analysis while another in social relationship examination) and reach for collaborators when other knowledge is needed. In such case, common knowledge in shared entities would actually be redundant. Individuals do not need to access content of entities created by others, but only keep aware of the creator.

### DISCUSSION

The goal of the study is to explore design opportunities to support collaboration in intelligence analysis by evaluating tool usage in a natural environment over multiple usage sessions. Our work builds upon prior empirical studies (e.g. [5, 4, 22, 8]) and embodies their design implications in our tool. The study also complements research that only tests tools in short

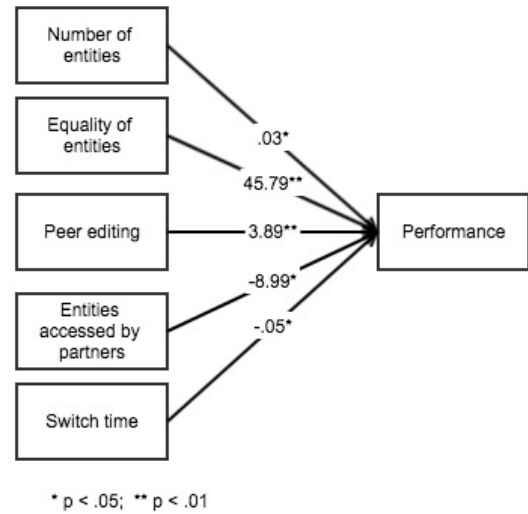


Figure 9. The relationship between collaboration characteristics and team performance.

term lab studies (e.g. [9, 13, 15]) by investigating tool usage over multiple usage sessions.

The study provides encouraging results on supporting collaboration in intelligence analysis. Participants appreciated an all-in-one environment where they could share raw documents, evidence snippets, views of evidence and hypotheses in one place. They liked the fact that they could contribute simultaneously without blocking or interfering each other. Another benefit of the collaborative tool is to keep teammates aware of each other's activities. Staying aware of teammates not only helps establish a common ground for planning team strategy, but also ensure everyone is following the plan as expected. Moreover, results suggest the awareness features provide positive *social facilitation* [39]: individuals found the task motivating and engaging with awareness of each other's activity. We also measured collaboration characteristics that impacted team performance, and found that balanced contribution, peer editing, and earlier switching from modeling to analysis predicted higher team performance.

#### Analyst-centered design

A critical requirement of developing tools that meet user needs is to understand their needs and practices. When these needs and practices are specialized (as is the case in Intelligence Analysis), it is particularly important to include the target user population in the design process. Intelligence analysis is a specialized, domain-specific task and it is important to understand analysts with domain knowledge, learn their practice, and observe their react to tool features [31]. However, professional analysts are very hard to include in a long-term design cycle due to confidentiality and security issues. Our classroom study provided an opportunity with deep access to analysts in training. These analysts have been trained with knowledge and skills of intelligence analysis, and have experience with state-of-the-art analytic tools such as Analyst's Notebook and

PARC ACH. In their reflections, participants often compared CAnalytics to those tools, as well as the different teamwork experiences. Therefore, while their feedback is admittedly not the same as an experienced professional, their feedback does provide a deeper insight into strengths and weaknesses of CAnalytics. In addition, the students are young learners that are willing to employ new work practices supported by features in tools. They are important parts of the future intelligence community. In some sense, their practice can be treated as a view into the future of practice of the community [25].

Our study spanned multiple usage sessions. This allowed participants time to learn to adapt to team functions and to appropriate the tool to best serve their team purpose. Teams were able to explore different team strategies, and to make changes if they got stuck [32]. For example, we noticed that two teams decided to change the use of the tool halfway in their analysis. One team started with dividing work by case documents, but later decided members should annotate different entity types. Another team started with an accretion strategy by annotating all entities. Later they discovered that this strategy brought too much noise, and decided to clean out irrelevant entities (filtering strategy). Such change occurs as a consequence of increased awareness of team functions and tool capabilities, which takes time to develop.

With deep access to these analysts, we are able to interpret and triangulate our data from multiple sources. We qualitatively analyzed participants' feedback to understand their first person experience, and corroborated them with surveys and interaction logs. The interaction logs provided us a detailed view into their analysis process, and allowed us to quantitatively characterize their team behavior. Finally, their analytic products, including the visual artifacts and team reports, helped us distinguish team outcomes.

Yet classroom study also has limits. For example, many factors and variables could exist that affect team performance. The fact that these factors are often impossible to model or control adds to the difficulty in data analysis (e.g. teammate absence). Also, data collection is challenging because team interactions are not always accessible. Teams can choose to work synchronously or asynchronously, and it is difficult to predict when or where the interaction of most interest is to occur. Verbal communication is not accessible, which could be useful to infer team awareness as a complement to interaction logs. Our work identifies both positive evidence and problematic situations, and propose potential solutions and possible hypotheses, yet rigorously evaluating these solutions and validating hypotheses is beyond this study. Lab studies and case studies can be conducted in the future to address these problems with greater control and deeper data access.

#### *Provide nuanced awareness for close collaboration*

Our study suggests that tools play an important role in shaping, or at least supporting, user's behavior towards more collaborative behavior. With traditional single-user tools, students often employ a divide-and-conquer strategy; they divide their job by tools, work individually on separate tools, and compile the results together in the end. In our study, we observed many teams spontaneously conducted closer collaboration and

enjoyed being able to contribute simultaneously. A potential problem with simultaneous contribution is duplicated or conflicted efforts. To minimize the problem, further nuanced awareness information of partner's action could be added. For example, Hajizadeh et al. [15] revealed collaborator's action in a transparent color. We can visualize where the collaborator is making an annotation as an indicator of interest of that piece of text. This is like typing indicator bubbles in chatting tools (e.g. Facebook Messenger) which provides awareness of actions currently being performed before outcome becomes visible.

#### *Enable an interleaving workflow*

A misconception about information analysis is that data modeling and data analysis are two staged activities. This is akin to the *waterfall* software development model, which features a sequential process that flows downwards through the phases of requirement conception, software design, software implementation, testing and maintenance. Critics have pointed out that the staged approach may not work properly, and an iterative design process is often required that leads to reframe user requirements, redesign, redevelopment, and retesting.

Our result demonstrates a similar iterative and dynamic process in intelligence analysis. The result is striking especially because our participants have been trained with tools that impose a waterfall model. They could have followed their old waterfall practice with our tool, but instead all teams spontaneously switched to an iterative manner. Indeed, relying only on information that has already been modeled and delivered to analysts will probably not solve all analytical problems [19]. It will probably be necessary to look elsewhere than evidence already extracted, re-model the data, and dig for more information. Re-modeling of the data could lead to a totally different picture of a problem (e.g. adding a link between two clusters changes the layout of the network, and thus framing of relationships between two robberies), leading to another analytic path.

#### *Design a richer graphic language for uncertainty*

We noted the importance of representing uncertainty. We observed teams in our study spontaneously employed two different approaches to deal with team uncertainty: either to mix them for better synthesis or to separate them for better clarification. We propose that a richer graphic language and interaction be designed so that analysts can encode uncertainty into the network view. For example, links and entities with different uncertainty can be visualized in different transparency. Users can *filter* by uncertainty so that users can choose to have only facts to take into account or review all inferences.

#### *Make valuable contributions more visible*

We noted cases where teams created far more entities than needed with an accretion strategy. Strikingly, while similar data modeling strategy was reported in the paper prototype study [5], users with CAnalytics seemed far more tempted to accretively add information, with far more entities and cluttered views. For example, the extreme team created as many as three times entities than the rest teams in our study, much more than difference in the paper prototype study. Why did this happen? We guess both the context of classroom study

and the system design contributed. Unlike in the lab study where teams are temporarily assembled, teams in a class evaluate peers either consciously or unconsciously and value how themselves are being evaluated. Such social pressure motivate individuals to make contributions, and indeed to make *visible* contributions, more than valuable contributions. That is, participants noticed that their work activity was visible to their partners, and accordingly prioritized doing more visible work over doing less visible work. In some cases, this led to a new problem of easy and less valuable contributions that were highly visible - such as creating and therefore sharing data entities that were not particularly important, and subsequently made data models seem cluttered. For example, creating and therefore sharing an entity gets immediately notified to the team whereas weighing the importance and relevance of an entity goes silent in the system. We need to investigate approaches to making significant contributions more visible, or perhaps making it more immediately visible that less important contributions are indeed less important.

#### *Build collapsible views for multi-level analysis*

Views could still get cluttered as data volume increases and analysts dig into event details (e.g. representing suspect's all actions to identify patterns of common actions in two robberies). Indeed, analysts often engage in multi-level analysis in parallel, frequently coordinating among, say, confirmation of location of an event, to comparison of two events, to overviewing all events as a robbery as a whole. A possible design solution is to enable collapsible data views. A collapsible view can help analysts focus attention on a certain level of details at a time while conveniently switching between levels, and when in collaboration, draw teammate's attention to a specific item.

#### *Share views as team resource*

An important improvement to make is to support view sharing, or sharing of intermediate analytic status. A common solution is to have a distinction of public view and private view [9, 14]. However, a nuanced difference exists in intelligence analysis: a view is not useful if it does not demonstrate an insight. A public view is not always necessary, and when analysts do obtain an insight, they may share multiple views to support and contextualize the insight. We propose that views should be treated as a team *resource*, just like data. Views as resource are sharable, extensible, and reusable. For example, an analyst can deliberately save their view as a shared resource when they feel it appeals to collaborators. Other people can reuse the view to their need. Shared views should be interactive rather than static images, so that analysts can perform all interactions including filtering and highlighting, and are able to evolve the view with collective team efforts, a critical requirement emphasized in [5].

## CONCLUSION

In this paper, we present findings from a classroom study in which teams of intelligence analysts in training collaboratively completed a complex intelligence project mediated by our tool. Our study provides encouraging results on supporting team interactions. Analysts engaged in an interleaving workflow and enjoyed a collaboration analytic process. We also identify new problems, such as data uncertainty, contribution visibility,

and intermediate insight sharing. We propose design implications to address these problems and will investigate these approaches in our future iterative design and evaluation.

While this study was performed with an emphasis on intelligence analysis, we argue that CANalytics can also be utilized for a broader domain of collaborative information analysis, wherein a team examines a complex information space of facts and relationships. These examples include business intelligence, scientific research, and social constructivist learning, in addition to intelligence analysis. As collaborative information analysis is increasingly a typical and chronic task, it is important for research to examine, understand, and provide effective tools and environments for these long-term, real-world CSCW interactions. This requires situating research in more complex work activity contexts, and directly investigating interactions, experiences, and outcomes in those contexts.

## ACKNOWLEDGMENTS

To be added

## REFERENCES

1. 2008. *Vision 2015: A Globally Networked and Integrated Intelligence Enterprise*. Technical Report. DOI: <http://dx.doi.org/10.1017/CB09781107415324.004>
2. Richard V Badalamente and Frank L Greitzer. 2005. Top Ten Needs for Intelligence Analysis Tool Development. *First Annual Conference on Intelligence Analysis Methods and Tools* May (2005).
3. Eric a Bier, Stuart K Card, and John W Bodnar. 2010. Principles and tools for collaborative entity-based intelligence analysis. *IEEE transactions on visualization and computer graphics* 16, 2 (2010), 178–91. DOI: <http://dx.doi.org/10.1109/TVCG.2009.104>
4. Marcela Borge, Craig H. Ganoe, Shin-I Shih, and John M. Carroll. 2012. Patterns of team processes and breakdowns in information analysis tasks. In *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work - CSCW '12*. ACM Press, New York, New York, USA, 1105–1114. DOI: <http://dx.doi.org/10.1145/2145204.2145369>
5. John M. Carroll, Marcela Borge, and SI Shih. 2013. Cognitive artifacts as a window on design. *Journal of Visual Languages & Computing* 24, 4 (2013), 248–261. <http://www.sciencedirect.com/science/article/pii/S1045926X13000207>
6. John M. Carroll, Dennis C. Neale, Philip L. Isenhour, Mary Beth Rosson, and D.Scott McCrickard. 2003. Notification and awareness: synchronizing task-oriented collaborative activity. *International Journal of Human-Computer Studies* 58, 5 (may 2003), 605–632. DOI: [http://dx.doi.org/10.1016/S1071-5819\(03\)00024-7](http://dx.doi.org/10.1016/S1071-5819(03)00024-7)
7. John M. Carroll, Mary Beth Rosson, Gregorio Convertino, and Craig H. Ganoe. 2006. Awareness and teamwork in computer-supported collaborations. *Interacting with Computers* 18, 1 (jan 2006), 21–46. DOI: <http://dx.doi.org/10.1016/j.intcom.2005.05.005>

8. George Chin Jr, Olga A. Kuchar, and Katherine E. Wolf. 2009. Exploring the analytical processes of intelligence analysts. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 11–20. <http://dl.acm.org/citation.cfm?id=1518704>
9. Gregorio Convertino, Helena M. Mentis, Aleksandra Slavkovic, Mary Beth Rosson, and John M. Carroll. 2011. Supporting common ground and awareness in emergency management planning. *ACM Transactions on Computer-Human Interaction* 18, 4 (dec 2011), 1–34. DOI: <http://dx.doi.org/10.1145/2063231.2063236>
10. Michael Diehl and Wolfgang Strpebe. 1987. Productivity Loss In Brainstorming Groups : Toward the Solution of a Riddle. *Journal of personality and social psychology* 53, 3 (1987), 497–509. DOI: <http://dx.doi.org/10.1037/0022-3514.53.3.497>
11. Globalytica. 2017. Te@mACH. (2017). <http://www.globalytica.com/>
12. David Gotz and Michelle X Zhou. 2009. Characterizing users' visual analytic activity for insight provenance. *Information Visualization* 8, 1 (2009), 42–55. DOI: <http://dx.doi.org/10.1057/ivs.2008.31>
13. Nitesh Goyal and Susan R Fussell. 2016. Effects of Sensemaking Translucence on Distributed Collaborative Analysis. In *CSCW '16 Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*.
14. Saul Greenberg. 1990. Sharing views and interactions with single-user applications. *COCIS '90 Proceedings of the ACM SIGOIS and IEEE CS TC-OA conference on Office information systems* 11, 2-3 (1990), 227–237. DOI: <http://dx.doi.org/10.1145/91478.91546>
15. Amir Hossein Hajizadeh, Melanie Tory, and Rock Leung. 2013. Supporting awareness through collaborative brushing and linking of tabular data. *IEEE transactions on visualization and computer graphics* 19, 12 (dec 2013), 2189–97. DOI: <http://dx.doi.org/10.1109/TVCG.2013.197>
16. Sandra G. Hart and Lowell E. Staveland. 1988. Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. *Advances in Psychology* 52, C (1988), 139–183. DOI: [http://dx.doi.org/10.1016/S0166-4115\(08\)62386-9](http://dx.doi.org/10.1016/S0166-4115(08)62386-9)
17. Linda D. Henman. 2003. Groups as Systems. In *Small group communication theory & practice: An anthology* (8th ed.), Randy Y. Hirokawa, S. Robert, Larry A. Samovar, and Linda A. Henman (Eds.). Oxford University Press, New York, 3–7.
18. Thomas Herrmann, Alexander Nolte, and Michael Prilla. 2013. Awareness support for combining individual and collaborative process design in co-located meetings. *Computer Supported Cooperative Work* 22, 2-3 (2013), 241–270. DOI: <http://dx.doi.org/10.1007/s10606-012-9179-x>
19. Richards J Heuer. 1999. *Psychology of intelligence analysis*. Lulu.com.
20. IBM. 2017. i2 Analyst's Notebook. (2017). <http://www-03.ibm.com/software/products/en/analysts-notebook/>
21. Petra Isenberg, Anthony Tang, and Sheelagh Carpendale. 2008. An exploratory study of visual information analysis. *Proceeding of the twenty-sixth annual CHI conference on Human factors in computing systems - CHI '08* (2008), 1217. DOI: <http://dx.doi.org/10.1145/1357054.1357245>
22. Youn-ah Kang and John Stasko. 2011. Characterizing the intelligence analysis process: Informing visual analytics design through a longitudinal field study. In *2011 IEEE Conference on Visual Analytics Science and Technology (VAST)*. Ieee, 21–30. DOI: <http://dx.doi.org/10.1109/VAST.2011.6102438>
23. Aleksei N Leont'ev. 1974. The problem of activity in psychology. *Soviet psychology* 13, 2 (1974), 4–33.
24. Narges Mahyar and Melanie Tory. 2013. Supporting Communication and Coordination in Collaborative Sensemaking. *Visualization and Computer Graphics, IEEE Transactions on* 20, 12 (2013), 1633 – 1642.
25. Iulian Martin. 2014. Current training approaches in intelligence analysis. In *The International Annual Scientific Session Strategies XXI* 2.
26. Judith S Olson, Dakuo Wang, Gary M Olson, and Jingwen Zhang. 2017. How People Write Together Now: Beginning the Investigation with Advanced Undergraduates in a Project Course. *ACM Transactions on Computer-Human Interaction (TOCHI)* 24, 4 (2017), 1–40. DOI: <http://dx.doi.org/10.1145/3038919>
27. PARC. 2017. ACH. (2017). <http://www2.parc.com/istl/projects/ach/ach.html>
28. Peter Pirolli and Stuart Card. 2005. The Sensemaking Process and Leverage Points for Analyst Technology as Identified Through Cognitive Task Analysis. In *Proceedings of International Conference on Intelligence Analysis*. 2–4. <http://vadl.cc.gatech.edu/documents/2>
29. R Core Team. 2016. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. <https://www.r-project.org/>
30. Kjeld Schmidt. 2002. The Problem with 'Awareness'. In *Computer Supported Cooperative Work (CSCW)*. 285–298.
31. Jean Scholtz and Alex Endert. 2014. User-centered design guidelines for collaborative software for intelligence analysis. *2014 International Conference on Collaboration Technologies and Systems (CTS)* (2014), 478–482. DOI: <http://dx.doi.org/10.1109/CTS.2014.6867610>
32. Gerry Stahl. 2006. *Group cognition: Computer support for building collaborative knowledge*. Mit Press Cambridge, MA.



33. John Stasko, Carsten Görg, and Robert Spence. 2008. Jigsaw: supporting investigative analysis through interactive visualization. *Information Visualization* 7, 2 (2008), 118–132. DOI: <http://dx.doi.org/10.1057/palgrave.ivs.9500180>
34. Gregory F Treverton. 2016. *New Tools for Collaboration: The experience of the U.S. Intelligence Community*. Number January. 1–44 pages.
35. United States Intelligence Community. 2017. Intelink. (2017). <https://www.intelink.gov>
36. Colin Ware. 2012. *Information visualization: perception for design*. Elsevier.
37. Daniel M Wegner. 1987. Transactive memory: A contemporary analysis of the group mind. In *Theories of group behavior*. Springer, 185–208.
38. K Wheaton. 2011. Wikis in intelligence. *Unpublished manuscript* (2011).
39. Robert Boleslaw Zajonc. 1965. Social facilitation. *Science, New Series* 149, 3681 (1965), 269–274.