Data Mining Meets the Needs of Disaster Information Management

Li Zheng, Chao Shen, Liang Tang, Chunqiu Zeng, Tao Li, Steve Luis, and Shu-Ching Chen, *Senior Member, IEEE*

***Abstract*—Techniques to efficiently discover, collect, organize, search, and disseminate real-time disaster information have become national priorities for efficient crisis management and disaster recovery tasks. We have developed techniques to facilitate information sharing and collaboration between both private and public sector participants for major disaster recovery planning and management. We have designed and implemented two parallel systems: a web-based prototype of a Business Continuity Information Network system and an All-Hazard Disaster Situation Browser system that run on mobile devices. Data mining and information retrieval techniques help impacted communities better understand the current disaster situation and how the community is recovering. Specifically, information extraction integrates the input data from different sources; report summarization techniques generate brief reviews from a large collection of reports at different granularities; probabilistic models support dynamically generating query forms and information dashboard based on user feedback; and community generation and user recommendation techniques are adapted to help users identify potential contacts for report sharing and community organization. User studies with more than 200 participants from EOC personnel and companies demonstrate that our systems are very useful to gain insights about the disaster situation and for making decisions.**

***Index Terms*—Data mining, disaster information management, dynamic query form, hierarchical summarization, user recommendation.**

I. Introduction

BUSINESS closures caused by disasters can cause millions of dollars in lost productivity and revenue. A study in Contingency Planning and Management shows that 40% of companies that were shut down by a disaster for three days failed within 36 months. Thin margins and a lack of a well-designed and regularly tested disaster plan can make companies, particularly small businesses, especially vulnerable [1]. We believe that the solution to better disaster planning and recovery is one where the public and private sectors work together to apply computing tools to deliver the right information to the right people at the right time to facilitate the work of those working to restore a community's sense of normalcy. While improved predictive atmospheric and hydrological models and higher quality of building materials and building codes are being developed, more research is also necessary for how to collect, manage, find, and present disaster information in the context of disaster management phases: preparation, response, recovery, and mitigation [4], [30].

In the United States, the Federal Emergency Management Agency (FEMA) has recognized the importance of the private sector as a partner in addressing regional disasters. The State of Florida Division of Emergency Management has created a Business and Industry Emergency Support Function designed to facilitate logistical and relief missions in affected areas. Four counties, Palm Beach, Broward, Miami-Dade, and Monroe, which constitute the Southeastern population of South Florida and include over 200 000 business interests, are developing Business Recovery Programs to help facilitate faster business community recovery through information sharing and collaboration.

Disaster management researchers at Florida International University have collaborated with the Miami-Dade Emergency Operations Center (EOC), South Florida Emergency Management and industry partners including Wal-Mart, Office Depot, Wachovia, T-Mobile, Ryder Systems, and IBM to understand how South Florida public and private sector entities manage and exchange information in a disaster situation. The efficiency of sharing and management of information plays an important role in the business recovery in a disaster [3]. Users are eager to find valuable information to help them understand the current disaster situation and recovery status. The community participants (the disaster management officials, industry representatives, and utility agents) are trying to collaborate to exchange critical information, evaluate the damage, and make a sound recovery plan. For example, it is critical that companies receive information about their facilities, supply chain, and city infrastructure. They seek this information from media outlets like television/radio newscasts, employee reports, and conversations with other companies with which they have a relationship. With so many sources of information, with different levels of redundancy and accuracy, possibly generated by a variety of reports (structured and unstructured), it is difficult for companies to quickly assimilate such data and understand their situation.

We have learned that a large-scale regional disaster may cause a disruption in the normal information flow, which in turn affects the relationships between information producers and consumers. Effective communication is critical in a crisis situation.

Manuscript received February 28, 2013; revised April 4, 2013, June 1, 2013, and July 10, 2013; accepted August 12, 2013. Date of publication October 4, 2013; date of current version October 16, 2013. The work was supported in part by the National Science Foundation under grants HRD-0833093, CNS-1126619, and IIS-1213026, the U.S. Department of Homeland Security under grant Award Number 2010-ST-062–000039, and Army Research Office under grant number W911NF-10–1–0366 and W911NF-12-1-0431. This paper was recommended by Associate Editor S. Rubin.

The authors are with the School of Computing and Information Sciences, Florida International University, Miami, FL 33199 USA (e-mail: lzhen001@cs.fiu.edu; cshen001@cs.fiu.edu; ltang002@cs.fiu.edu; czeng001@cs.fiu.edu; taoli@cs.fiu.edu; luiss@cs.fiu.edu; chens@cs.fiu.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/THMS.2013.2281762

What is not very well known is how to effectively discover, collect, organize, search, and disseminate real-time disaster information.

Our study of the hurricane disaster information management domain has revealed two interesting yet crucial information management issues that may present similar challenges in other disaster management domains. The first issue is that reconstructing or creating information flow becomes intractable in domains where the stability of information networks is fragile and can change frequently. However, important information networks often carry and store critical information between parties, which dominates the flow of resources and information exchanges. The consequence is that the ability and the efficiency of communication degrade once critical networks are disrupted by the disaster and people may not have alternative paths to transfer information. For example, once power is disabled and uninterruptable power supplies fail after a hurricane, computing and networking equipment will fail unless preventative measures are taken. However, maintaining a fuel-consuming generator is not always possible.

Another issue is the large volume of disaster situation information. Reading and assimilating situational information are very time consuming and may involve redundant information. Thus, to quickly reassemble or create information flows for multiparty coordination activities during disaster situations, technologies that are capable of extracting information from recent updates, delivering that information without conflict or irrelevance, and representing preferential information are needed.

This research is mainly focused on the second issue. Research in disaster management addresses the needs and challenges of information management and decision making in disaster situations [36]–[38]. We have developed an understanding of those needs for hurricane scenarios. The information delivery should support users' complex information needs tailored to the situation and the tasks; and the information should be synthesized from heterogeneous sources and tailored to specific contexts or tasks at hand. It should be summarized for effective delivery and immediate usefulness for making decision.

A. Related Work

The approaches and the tools that are used for information sharing vary based on the task and scale of the participating agencies or the types of information exploration platforms.

Commercial systems, such as WebEOC [39] and E-Teams [40] used by Emergency Management departments located in urban areas, can access multiple resources. A Disaster Management Information System developed by the Department of Homeland Security is available to county emergency management offices and participating agencies to provide an effective reports/document sharing software system. The National Emergency Management Network [41] allows local government to share resources and information about disaster needs; The RESCUE Disaster Portal is a web portal for emergency management and disseminating disaster information to the public [4]; The Puerto Rico Disaster Decision Support Tool is an Internet-based tool for disaster planners, responders, and related officials at the municipal, zone, and state level for access to a variety of geo-referenced information [35].

Efforts, such as GeoVISTA [31], facilitate the information distribution process in disasters. GeoVISTA monitors tweets to form situation alerts on a map-based user interface according to the geo-location associated with the tweets. Such a system applies geographic information sciences to scientific, social, and environmental problems by analyzing geospatial data [31].

These useful situation-specific tools provide query interfaces, and GIS and visualization capabilities to simplify the users' interaction and convey relevant information. The primary goal of these systems are message routing, resource tracking, and document management for the purpose to support situation awareness, demonstrate limited capabilities for automated aggregation, data analysis, and mining [4].

However, these tools do not consider how different communities interact with other businesses and county organizations. Further, these tools do not allow for the integration of real-time information. They do not provide information extraction (IE), information retrieval (IR), information filtering (IF), and data mining (DM) techniques needed when delivering personalized situation information to different types of users.

B. Design Challenges

We have identified four key design challenges for disaster information sharing platforms and tools.

1) *Effective techniques to capture the status information:* Participants need to communicate status through many channels, including email, mailing lists, web pages, press releases, and conference calls. It is desirable to capture such status information when it is available and to prevent redundant reporting. To facilitate the reuse of such materials, users should be able to update status information via unstructured documents such as plain text, Adobe PDFs, and documents. It is necessary to identify the useful information in the documents.

2) *Effective and interactive information summarization methods:* It is important to build a summarized view to support understanding the situation from reports. Multidocument summarization provides users with a tool to effectively extract important and related ideas of current situations. Previous text summarization techniques gave users a fixed set of sentences based on the user query. An interactive summarization interface is needed to help users navigate collected information at different granularities, and locate their target information more efficiently.

3) *Intelligent information delivery techniques:* Data can be collected through different channels and may belong to different categories. During disaster preparation and recovery, users do not have the time to go through the system to find the information they want. Structured information can help people make decisions by providing them with actionable and concrete information representation and exploration. However, navigating large datasets on a mobile device is particularly inefficient. An interactive tabular interface can help users filter useful information by adaptively changing query conditions and user feedback.

4) *Dynamic community generation techniques:* In information sharing tasks, identifying a group of recipients to which a certain type of information is conveyed can improve the efficiency of communication. In addition, identifying how participants interact with these communities in a disaster situation may reveal information helpful in a recovery scenario. User recommendation techniques can automatically and interactively generate potential recipients for different pieces of information. In addition, user recommendation techniques can help to dynamically organize user groups according to various information sharing tasks.

We created an information-rich service on both web-based and mobile platforms in the disaster management domain to address the design challenges. In particular, to address the first challenge, we apply information extraction to automatically extract the status information from documents. To address the second challenge, we apply hierarchical summarization to automatically extract the status information from a large document set and also provide a hierarchical view to help users browse information at different granularities. To address the third, we create a user interface capability called the dynamic dashboard to improve information quality to match user's interests, and use document summarization techniques to give users fast access to multiple reports. In addition, a dynamic query form is designed to improve information exploration quality on mobile platforms. It captures users' interests by interactively allowing them to refine and update their queries. To address the fourth challenge, for community discovery, we adopt spatial clustering techniques to track assets like facilities, or equipment, which are important to participants. The geo-location of such participants can be organized into dynamic communities, and these communities can be informed about events or activities relevant to their spatial footprints. For user recommendation, we use transactional recommendation history combined with textual content to explore the implicit relationship among users.

Thus, we designed and implemented a web-based prototype of a Business Continuity Information Network (BCiN) that is able to link participating companies into a community network, provide businesses with effective and timely disaster recovery information, and facilitate collaboration and information exchange with other businesses and government agencies. We also designed and implemented an All-Hazard Disaster Situation Browser (ADSB) system that runs on Apple's mobile operating system (iOS), and iPhone and iPad mobile devices. Both systems utilize the data processing power of advanced information technologies for disaster planning and recovery under hurricane scenarios. They can help people discover, collect, organize, search, and disseminate real-time disaster information [4], [5].

This study introduces a unified framework that systematically integrates the different techniques developed in [5] and [29]. The idea is that such a framework can be utilized when dealing with different systems or applications separately (e.g., BCiN and ADSB), and hopefully can be easily applied to other scenarios having critical information sharing and management needs.

The rest of the paper is organized as follows. Section II describes the system architecture: information extraction techniques to create structured records, the hierarchical summarization module, the dynamic dashboard and the dynamic query form modules, and community identification and user recommendation modules. Section III presents two case studies of the BCiN and ADSB systems. Section IV describes the system evaluation and data crawling strategies. The conclusion is in Section V.

TABLE I
EXAMPLE OF EOC REPORT

Time: October 21, 2005 12:30 p.m.
Miami-Dade Emergency Operations Center is currently activated at a level II and officials and emergency managers are carefully monitoring Hurricane Wilma.
Residents are urged to finalize their personal hurricane preparations.
On Monday, October 24, Miami-Dade County offices, public schools, and courts will be closed.
Currently, transit bus and rail service continues, including Metrobus, Metrorail and Metromover.
Miami International Airport is open. However, if you have travel plans please check with your airline for flight information.
Tomorrow afternoon, the American Red Cross will open hurricane evacuation centers for residents who do not feel safe in their homes or live in low-lying areas.

II. SYSTEM ARCHITECTURE

A. Structured Information Extraction From Reports

A user interface supports information sharing among companies and government agencies. We do not request a unified format for them to submit the reports. Instead, we use information extraction methods to integrate reports from different sources. For example, Table I shows an example of EOC reports.

The key information is "What was/is/will be the status of Facilities/Services/... at the time of ...." From the EOC reports, we need to extract such information in the form of a triple, entity, time, and status, which reveals the status information of the entity at a certain time. In EOC reports, the entity may be a facility or public service like "Miami International Airport," "schools," "bus," and an order like "curfew." If the entity represents an order, the triple means whether the order is in effect (or not) at that specific time. We extract these triples through two steps: first, we extract entities and time expressions, then, we classify a pair of (service, time) to a proper category, "no relation"/"open"/"close"/"unclear." We assume that the information of one event is described in one sentence, so we process every sentence individually to extract an event. To extract those triples, both entity and relation extraction will be performed. Sometimes two different reports generate the same events, which have the same extracted information, such as the same hurricane name, the same date and the same status of traffic. The repeated events will be deleted. Note that the date/time is an important attribute for every event. Two events with different date/time (at the hourly level) are treated as two different events.

1) Entity Extraction: For each report, sentence segmentation is conducted, and each sentence is Part-Of-Speech-tagged [34]. To extract entities and time expressions, we manually label some news and train a linear chain conditional random fields model to

TABLE II
ENTITY EXTRACTION RESULT OF THE REPORT IN TABLE I

Miami-Dade Emergency Operations Center is currently activated at a level II and officials and emergency managers are carefully monitoring Hurricane Wilma.
Residents are urged to finalize their personal hurricane preparations.
On <T>Monday, October 24</T>, <E>Miami-Dade County offices</E>, <E>public schools</E>, and <E>courts</E> will be closed.
<T>Currently</T>, <E>transit bus</E> and <E>rail service</E> continues, including <E>Metrobus</E>, <E>Metrorail</E> and <E>Metromover</E>.
<E>Miami International Airport</E> is open. However, if you have travel plans please check with your airline for flight information.
<T>Tomorrow afternoon</T>, the American Red Cross will open <E>hurricane evacuation centers</E> for residents who do not feel safe in their homes or live in low-lying areas.

tag all words, using "BIO" annotation [6], [7]. A word tagged as [TYPE-B]/[TYPE-I] means it is the beginning/continuing word of the phrase of the TYPE, and the word tagged by O means it is not in any phrase. TYPE can be E for entity or T for time expression. Given sentence X, the probability that its tags are Y is

$$p(Y|X) = \frac{1}{Z_X} \exp\left(\sum_{i,k} \lambda_k f_k(y_{i-1}, y_i, X) + \sum_{i,l} \mu_l g_l(y_{i-1}, y_i, X)\right) \quad (1)$$

where Z_X is the normalization constant that makes the probability of all state sequences sum to one; $f_k(y_{i-1}, y_i, X)$ is an arbitrary feature function over the entire observation sequence and the states at positions i and $i-1$, while $g_l(y_{i-1}, y_i, X)$ is a feature function of the states at position i and the observation sequence; and λ_k and μ_l are the weights learned for the feature functions f_k and g_l, reflecting the confidence of feature functions by maximum likelihood procedure. The most probable labels can be obtained as

$$Y^* = \underset{Y}{\operatorname{argmax}}\, P(Y|X) \quad (2)$$

by the Viterbi-like dynamic programming algorithm [6]. Features that we use are the local lexicons and POS tags, and the dictionary composed of the existent entity names in the database. Table II shows the entity extraction results of the report in Table I.

2) Relation Extraction: If a sentence contains an entity but no time expression, the time of the report will be associated with the sentence. To generate the triple by connecting the entity with the time expression with a proper status label, we train a multicategory support vector machine [8] to classify each pair of (entity, time) to a proper category, defined as "no relation"/"open"/"close"/"unclear." Table III shows the features that we used for classification, from which the TenseOfSentence(*e,t*), NegativeVerbsInSentence(*e,t*), and PositiveVerbsInsentence(*e,t*) are extracted as the heuristic rules to indicate the tense of the sentence, the verbs with negative modifiers, and the verbs without negative modifiers semantically in the sentence. Note that FromDocument(t) indicates whether the time is the time associated with document.

TABLE III
FEATURES USED TO CLASSIFY WHETHER THE ENTITY e IS ASSOCIATED WITH THE TIME EXPRESSION t

DistanceBetween (e, t)
WordBetween(e,t)
TenseOf Sentence(e,t)
NegativeVerbsInSentence(e,t)
PositiveVerbsInsentence(e,t)
ContainDate(t)
PrepositionBefore(t)
FromDocument(t)

TABLE IV
INFORMATION EXTRACTED FROM THE EOC REPORT IN TABLE I

Service	Time	Status
Miami-Dade County offices	October 24, 2005	close
public schools	October 24, 2005	close
courts	October 24, 2005	close
transit bus	October 22, 2005 6:30 p.m.	open
Rail service	October 22, 2005 6:30 p.m.	open
…		
Miami International Airport	October 22, 2005 6:30 p.m.	open
hurricane evacuation centers	October 23, 2005 afternoon	open

We extract those pairs of entity and time expressions in "open" and "close" categories to form the triple. The time expressions are formatted into an absolute form of expression from relative time expressions such as "next Monday," "this afternoon" using the time of the report as a benchmark. The structured information that is extracted from the report in Table I is shown in Table IV.

B. Report Summarization

The hierarchical multidocument summarization method generates the hierarchical summaries of reports. We use the affinity propagation (AP) [9] clustering method to build a hierarchical structure for sentences of related reports.

1) Affinity Propagation: The input of the AP algorithm is the sentence similarity graph defined as *G<V, E>*: V is the set of vertices with each vertex, called data point, representing a sentence. E is the set of edges. Let $s(i,k)$ be the similarity between two distinct points i and k, indicating how well data point k is suitable to be the exemplar of point i. Especially, $s(i,i)$ is the preference of a sentence i to be chosen as the exemplar. There are two kinds of messages passing between data points: responsibility and availability.

The responsibility $r(i,k)$ is computed as follows:

$$r(i,k) \leftarrow s(i,k) - \max_{\{k' \neq k\}} \{a(i,k') + s(i,k')\}. \quad (3)$$

The responsibility $r(i,k)$ is passing from i to candidate exemplar k. It reflects the accumulated evidence of how well point k is selected as the exemplar for point i against other candidate exemplars.

The availability $a(i,k)$ is computed as follows:

$$a(i,k) \leftarrow \min\{0, r(k,k) + \sum_{i' \in V, i' \notin \{i,k\}} \max\{0, r(i',k)\}\}. \tag{4}$$

The availability $a(i,k)$ is passing from the candidate exemplar k to point i, reflecting the accumulated evidence of how appropriate point i to choose point k as its exemplar, considering the support from other points that share point k as exemplar, whereas the responsibility updating lets all candidate exemplars compete for the ownership of a data point, the availability updating gathers evidence from data points to measure the goodness of each candidate exemplar.

The self-availability $a(k,k)$ is updated as follows:

$$a(k,k) \leftarrow \sum_{i' \in V, i' \neq k} \max\{0, r(i',k)\}. \tag{5}$$

This message reflects accumulated evidence of point k being an exemplar based on the received positive responsibilities from other points.

All availabilities are initialized to zero: $a(i,k) = 0$. After the updating converges, availabilities and responsibilities are combined to identify exemplars. For point i, its corresponding exemplar is obtained by maximizing the following expression:

$$k^* = \arg\max_k \{a(i,k) + r(i,k)\}. \tag{6}$$

We choose AP for the following reasons.

1) AP can find clusters with much lower error than other clustering methods, such as K-Means [9].
2) AP performs efficiently on sparse similarity graphs, which is the case of document space. The run time for iterations is linear with the number of edges in the graph.
3) AP takes a real number as input, called the preference for each data point. The preference quantifies the likelihood of it being chosen as an exemplar. Thus, prior and heuristic knowledge can be used to associate different sentences with different preferences.
4) AP identifies exemplars for each cluster or group that can be naturally used as the summary sentences for the cluster.

2) Hierarchical Summarization on Affinity Propagation: For the sentences in related reports,$\{s_1, s_2, \ldots, s_n\}$, we want to build a hierarchical clustering structure and use exemplars of clusters as the summary. Starting from all sentences, we recursively apply AP in an agglomerative way to find proper exemplars until the number of exemplars is small enough. We pick 20 as the number of exemplars which means 20 sentences will be selected from the document set as the summary. The preference for each sentence and similarity between sentences are used as the input of the AP algorithm.

3) Sentence Preference: We define the preference of sentence i to be chosen as an exemplar using the following scores.

1) $LanguageModelScoreL$: For sentence i, L_i is calculated as the logarithmic probability of sentence i using unigram model training on the reports $\{s_1, s_2, \ldots, s_n\}$. Generally, a shorter sentence that has more frequent words in the reports has a higher score.
2) *LexPageRankscore* P: LexPageRank proposed in [10] calculates the page rank score of sentences on the sentence similarity matrix. The score measures the prestige in sentence networks assuming that the sentences similar to many of the other sentences in a cluster are more prestigious with respect to the topic. Since the original LexPageRank can be interpreted as the probability in random walk theory, we use the logarithmic version to make it at the same scale with the language model score.
3) *FreshnessScore* F: Users are generally more interested in the latest information; we calculate the freshness score of sentence i based on the age of the document containing i as

$$F_i = e^{-a_i} \tag{7}$$

where a_i is the *age* in terms of the number of days the document contains the sentence i. Clearly, $F_i \in [0,1]$ decreases as the document age increases. Another property is that for two sentences from two documents with some age difference (e.g., 1 day), the difference of their freshness scores is large when both sentences are relatively new. Thus, it can better differentiate freshness for latest information.

Finally, the preference of s_i is the sum of the three feature scores with a scaling parameter:

$$s(i,i) = (L_i + D_i + F_i) \times e. \tag{8}$$

The parameter e is obtained by experimentally testing the clustering results and choosing the value that achieves the best clustering performance.

4) Sentence Similarity: Sentence similarity $s(i,j)$ indicates how well the data point with index j is suited to be the exemplar for data point i. In our case, it means how likely sentence i can be summarized by sentence j. If sentence i and sentence j have nonstop word overlaps, we calculate $s(i,j)$ by the log-likelihood of sentence i given that its exemplar is sentence j as follows:

$$s(i,j) = \log P(i|j). \tag{9}$$

To calculate the conditional probability, a unigram language model is trained on sentence j by using the Dirichlet smoothing [32]. Then, the probability of sentence i is calculated by using the language model.

C. Dynamic Dashboards and Dynamic Query Form

1) Dynamic Dashboard:

a) Challenges for dashboards: When a disaster happens, the system will receive a lot of information at once. It is necessary for the system to select a small portion of entities that a user really cares about to display in the dashboards. The dashboards provide condensed views for users to quickly explore the recent news and reports. It cannot display all the information in such a small area.

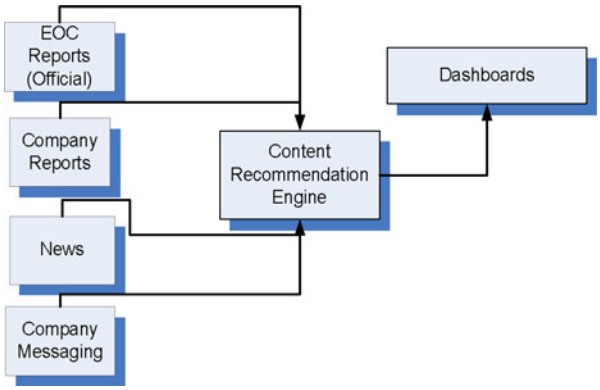


Fig. 1. Content recommendation engine.

Another problem in practice is that the information sent from company users may have a lot of redundancy. For instance, when a hurricane arrives in South Florida, almost all the company users in that area will report the same hurricane information: "*The storm has arrived South Florida*." Thus, different users may report the same information hundreds of times. Therefore, the system has to identify which information is redundant, and the redundant information should not appear in the dashboards.

We address these problems by introducing the dynamic dashboard supported by the content recommendation engine. The engine's main task is to extract the most important, relevant and nonredundant information about entities from news and reports.

b) Content recommendation engine: Fig. 1 shows the data flow related to the content recommendation engine. There are four main data sources: EOC reports, news, company reports, and company messages. Since reports and news may contain information about multiple entities, in content recommendation engine, each report or news is divided into several documents. Each document consists of a sentence that contain entity status information plus a context window (one previous and next sentence).

The content recommendation consists of two steps. The first step is text clustering, which is to cluster the same description of entities into one cluster. The second step is ranking the text by the relevance and presenting the top k items to the dashboards.

The content recommendation engine is based on unstructured text, while the situation dashboard, thread dashboard, and company dashboard display structured information. The four dashboards are denoted as Db_S(situation dashboard), Db_T(threat dashboard), Db_E(event dashboard), and Db_C(company dashboard). The maximum numbers of items allowed to show in the dashboards Db_S, Db_T, Db_E, and Db_C are denoted as size_S, size_T, size_E, size_C, respectively.

The content recommendation engine recommends information from different data sources to the four dashboards. Table V shows the relationship between the data sources and the four dashboards. Since the dashboards show the latest information, we use the last 48 h records and news as the input of the engine.

For any user u, the set of information submitted by u is denoted by $I(u)$ and the set of reports/news of which the details are viewed by u is denoted by *J(u)*. *u*'s profile is composed of *I(u)* and *J(u)*.

TABLE V
DATA SOURCES OF DIFFERENT DASHBOARD

Dash-board	EOC Reports	Company Reports	News	Company Messaging
Db_S	√		√	
Db_T	√			
Db_F	√	√	√	√
Db_C		√		√

c) Document clustering: Before performing clustering, we use term frequency–inverse document frequency (TF–IDF) transformation [11] to transform the text data (report, news, and so on) to the vectors. The similarity between two documents can be calculated by the cosine similarity [12].

We apply the K-Medoids [13] algorithm to cluster the documents. Note k is a user-defined parameter, which is determined by the managers of the system. It is also relevant to the number of items allowed to be displayed on the dashboards. We present the top five $(k = 5)$ items in the dashboards.

After clustering, each cluster contains the duplicated information about an entity and one document can be selected from a cluster to show the status of the entity. However, before that, we have to decide which cluster and which document should be selected.

d) Content ranking: For a specific user u, there are three priorities of the information. The three priorities from highest to lowest are EOC reports, company partner's information (messages received) and other users' information (company reports). The three priorities are denoted by user-defined parameters pr_1, pr_2, and pr_3, respectively, and $pr_1 > pr_2 > pr_3 > 0$. For a given document $d_i \in D$, we use $pr(d_i)$ to indicate the priority of this document, and $pr(d_i) \in \{pr_1, pr_2, pr_3\}$.

Suppose the current user is u, $t(u)$ represents the term vector representation of the documents submitted or read by *u* .We can obtain the u's feature f_u by users' profiles as follows:

$$f_u = \alpha \frac{\sum_{u \in I(u)} t(u)}{\left|\sum_{u \in I(u)} t(u)\right|} + (1-\alpha) \frac{\sum_{u \in J(u)} t(u)}{\left|\sum_{u \in J(u)} t(u)\right|}. \quad (10)$$

The parameter α is used to tune the importance weights of the reports submitted and viewed as the profile. α is set to 0.8 in our work.

The importance score of each document $d_i \in D$ is calculated as follows where $t(d_i)$ represents the term vector representation of document d_i:

$$\mathrm{score}(d_i) = \mathrm{sim}(f_u, t(d_i)) \cdot pr(d_i). \quad (11)$$

For each dashboard, we use a top-K query to greedily search the K highest scores' documents from its corresponding data sources, where $K \in \{\mathrm{size}_S, \mathrm{size}_T, \mathrm{size}_E, \mathrm{size}_C\}$ and no two documents are selected from the same cluster. The set of K highest scores' documents is the result of the content recommendation engine. The EOC official reports have the highest priority. Some of them are not very relevant to the current user; however, information from these reports is still likely to appear on the event dashboard.

2) Dynamic Query Form: Each report is associated with a set of attributes, such as the report location, date, or annotations added by the creator. Such structural information allows users to execute relational queries on reports. For example, we want to find those reports that are about hurricanes from 1990 to 2010 and the latitude of the hurricane center is above 30°. Hence, our system applies query forms for users to support relational queries.

Traditional query forms are statically embedded by developers or database administrators. Those static query forms are used for the static database schema. However, different reports have different sets of attributes. For example, the hurricane report and the earthquake report use two very distinct sets of attributes. Furthermore, the associated values of annotation attributes created by the user at runtime are inconsistent. Therefore, it is impossible to design a static and fixed query form to cover all those attributes. Therefore, we implement the dynamic query form to satisfy those dynamic and heterogeneous query desires.

Previous research on database query forms focuses on how to automatically generate the query form from the data distribution or query history [14]–[17]. However, different users can have different query desires. How to capture the current user's interests and construct appropriate query forms are the key challenges for query form generation that has not been solved.

a) Problem formulation: Query forms are designed to return the user's desired results. The metric of the goodness of a query form is based on two traditional measures of evaluating the quality of the query results: *precision* and *recall*.

Let $F = (\mathbf{A}_F, \sigma_F)$ be a query form with a set of query conditions σ_F and a set of displaying attributes $\mathbf{A}_F$. Let D be the set of all reports in the database; $|D|$ is the total number of reports. $P_u(.)$ is the distribution function of user interests; $P_u(d)$ is the user interest for a report d, and $P_u(A_F)$ is the user interest for an attribute subset A_F; and $P(\sigma_F|d)$ is the probability of query condition σ_F being satisfied by d, i.e., $P(\sigma_F|d) = 1$ if d is returned by F and $P(\sigma_F|d) = 0$ otherwise. Then, given a query form $F = (\mathbf{A}_F, \sigma_F)$, the *expected precision*, *expected recall*, and *expected fscore* of F are defined as follows:

$$\text{Precision}_E(F) = \frac{\sum_{d\in D} P_u(d)\, P_u(A_F)\, P(\sigma_F|d)}{\sum_{d\in D} P(\sigma_F|d)} \tag{12}$$

$$\text{Recall}_E(F) = \frac{\sum_{d\in D} P_u(d)\, P_u(A_F)\, P(\sigma_F|d)}{\sum_{d\in D} P_u(d)\, P_u(A)} \tag{13}$$

$$F\text{Score}_E(F) = \frac{\left(1+\beta^2\right)\cdot \text{Precision}_E(F)\cdot \text{Recall}_E(F)}{\beta^2\cdot \text{Precision}_E(F)\cdot \text{Recall}_E(F)} \tag{14}$$

where $A_F \subseteq A, \sigma_F \in \sigma$, and β is a parameter defined by the user and β is usually set to 2.

$F\text{Score}_E(.)$ is the metric to evaluate the overall goodness of a query form. The problem of our dynamic query form [43] is how to construct a query form $\hat{F}$ that maximizes the goodness metric $F\text{Score}_E(.)$, i.e.

$$\hat{F} = \underset{F}{\text{argmax}}\, F\text{Score}(F). \tag{15}$$

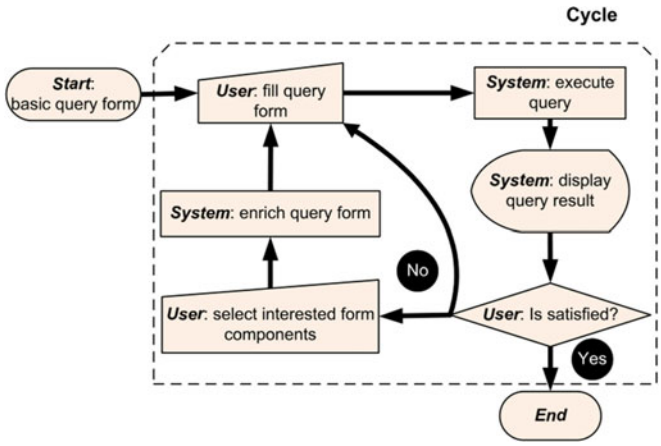


Fig. 2. Flowchart of dynamic query form.

b) Method description: It is impractical to construct an optimal query form $\hat{F}$ at the very beginning, since we do not know which reports and attributes are desired by the user. In other words, estimating $P_u(d)$ and $P_u(\mathbf{A}_F)$ is difficult.

The ADSB system provides an iterative way for the user to interactively enrich the query form. Fig. 2 shows the work flow of our dynamic query form system. At each iteration, ADSB computes a ranked list of query form components for users, and then, lets users make the choice for their query form. Those query form components are ranked by the metric $F_E(F)$.

There are two types of query form components: attribute display and query condition.

Assuming the current query form is F_i in the flowchart, and the next query form is F_{i+1}. We need to estimate $P_u(d)$, $P_u(\mathbf{A}_{F+1})$, and $P(\sigma_{F+1}|d)$ to compute $F\text{Score}_E(F_{i+1})$. The estimation is based on user behaviors when interacting with the ADSB system. Let $D_{u,f}$ be the set of reports viewed by the users and d' be one of the document in $D_{u,f}$. We assume those reports are interesting to the current user, then

$$P_u(d) = \sum_{d'\in D_{u,f}} P_u(d|d')\, P_u(d'). \tag{16}$$

We use the random walk model to compute the relevance score between reports as the value of $P_u(d/d')$ [18].

Suppose A is displaying an attribute we suggest for query form F_{i+1} and $\boldsymbol{A}_{f_{i+1}} = A \cup \boldsymbol{A}_{F_i}$, where $A \in \boldsymbol{A}, A \notin \boldsymbol{A}_{F_i}$. Therefore, $\boldsymbol{A}_{F_i}$ can be obtained in the current query form F_i.

$$P_u\left(\boldsymbol{A}_{F_{i+1}}\right) = P_u\left(A|\boldsymbol{A}_{F_i}\right) P_u\left(\boldsymbol{A}_{F_i}\right). \tag{17}$$

We also estimate $P_u\left(A|\boldsymbol{A}_{F_i}\right)$ by using a random walk model on the *attribute graph*. The nodes of the attribute graph are report attributes, and the edges are common reports. Therefore, the weight of edge *ij* is computed by how many reports use both the two attributes i and j.

Suppose s is a query condition we suggest for query form F_{i+1}. Therefore, $\sigma_{F_{i+1}} = s \wedge \sigma_{F_i}$, where s is a single query condition for attribute $A_s, A_s \in \boldsymbol{A}$. σ_{F_i} can be obtained in the current query form F_i. For each report $d \in D, P\left(\sigma_{F_{i+1}}|d\right) = P(s|d)\, P\left(\sigma_{F_i}|d\right)$. It is very time consuming to find the best s

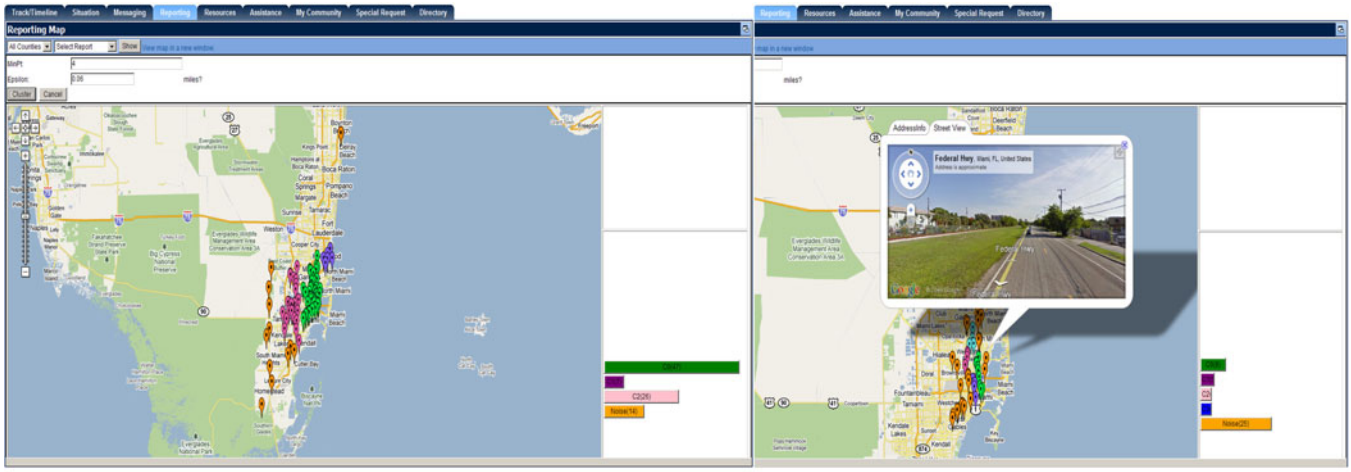


Fig. 3. Dynamic community generation result. (a) Generated communities. (b) Interactive clustering of large cluster.

by brute-force search on all $P(s|d)$. Therefore, we precompute the $P(s|d)$ and store it in the database.

D. Community Generation and User Recommendation

1) Community Generation: Two characteristics in disaster recovery scenarios motivate us to consider geo-location information. The first characteristic is that any event extracted from a report is associated with a/several location(s) indicating the place(s) where the announced event takes place. The second characteristic is that spatially colocated entities are more likely sharing similar disaster damage situations.

These two characteristics motivate the concept of community: a community is a certain geographical region in which entities tend to share more recovery status or interests in common. Therefore, geographically identifying those communities is important to help companies understand the current disaster situation and any interested resources nearby. Our system addresses community generation by adapting existing spatial clustering algorithms. In practice, we provide an interactive spatial clustering interface for users to access multilevel communities in a top–down manner and consider physical or nonphysical obstacles when generating spatial clusters to form more practical communities.

a) Spatial clustering: Spatial data clustering identifies clusters, or densely populated regions, according to some distance measurement in a large, multidimensional dataset [12], [13]. Many spatial clustering techniques [19], [20], [26] have been developed to identify clusters with arbitrary shapes of various densities and with different physical constraints.

In practice, communities formed by geographically related entities can be of various shapes. Therefore, we extend DBSCAN [19], a well-known density-based clustering algorithm, which is capable of identifying arbitrary shape of clusters, to generate dynamic communities.

b) Spatial clustering with constraints: We consider the method of spatial clustering with constraints. Generally, there are three types of constraints [13]. 1) Constraints on individual objects: such constraints are nonspatial instance-level constraints that can be preprocessed before performing clustering algorithms. 2) Constraints as clustering parameters: such constraints are usually confined to the algorithm itself. Usually, user-specified parameters are given through empirical studies. 3) Constraints as physical obstacles: such constraints are tightly intertwined with clustering process. It is clear that physical obstacles are such constraints that prevent two geographically close entities from being clustered together. For example, the bridge, highway and rivers are of this type.

In our BCIN system, we focus on object constraints and physical constraints.

Object constraints: We have two ways to obtain object constraints: 1) users submit formatted reports through report interface. Those reports are immediately recorded in the database; 2) our system extracts entity status from reports. For example, Table IV can be used as object constraints.

Obstacle constraints: A polygon is a typical structure in spatial analysis for modeling objects. Obstacles modeled by a polygon can be represented as a set of line segments after performing polygon reduction [20].

Fig. 3(a) shows the communities generated by clustering all open facilities and companies in Miami with the constraint: "I75 closed."

c) Interactive spatial clusters: In order to deal with unbalanced size of clusters, we provide users with an interactive mechanism to track the subcommunity information within a large size community. Further clustering process will be triggered in the runtime when a user selects a larger community and wants to see the cluster information within such a community at a finer granularity. By using this mechanism, users can obtain clusters with different granularities and more meaningful results. Fig. 3(b) shows the interactive clustering results within the largest cluster in Fig. 3(a).

2) User Recommendation: The user recommendation component provides an interface to explore other users' recommendations or share reports with other people. It also helps the user quickly identify sets of users with shared interests. It is designed

by considering each individual's transactional sharing history, textual content of each transaction and timeliness of interaction to provide each user with a personalized information sharing experience.

Related work has been applied to email communication networks analysis to find important persons, identifying frequent communication pattern and detecting communities based on transactional user relationships [21]–[25]. Those techniques can prevent a user from forgetting to add important recipients, avoid costly misunderstandings, and communication delays. Carvalho *et al*. [24] introduced several supervised learning models to predict the score of each user associated with a given email content. By aggregating TF–IDF vector of each email addressed to a user (by To, CC, or BCC), it can predict the score of a new email to such user. However, it was not aware of the different importance of emails for senders and recipients. Horn *et al*. [25] explicitly associated higher weights to senders, and also consider user-interaction graph as a directed hypergraph. It focused on the time and frequency of interactions but ignored the content information involved in each email, which could be an important indication of potential related users.

There are three practical considerations motivating the user recommendation: 1) to share information to the right/related people, users need an intelligent tool to autogenerate a recipient list that covers active users interested in specific information; 2) manually identifying meaningful groups of users is time consuming, therefore, users prefer efficient ways to organize contacts instead of navigating the contact list repeatedly; and 3) it could be more effective for a user to access information that others think is important.

Therefore, our system addresses the aforementioned issues by considering both user interactions and textual information. In practice, we provide dynamic user suggestions for the news recommendation and community recommendation interface to help our system users organize their critical partnerships.

a) Transactional interactions: An interaction or transaction is defined as the process of a user sharing a report with one or more other users. Therefore, the reports sharing transaction database can be treated as a hypergraph with each node representing a registered user and a set of edges created at the same time from one node to a set of nodes representing an occurred transaction. There are three important factors associated with each edge.

1) *Time:* The time that the transaction happened. It indicates the importance of recency. In general, the more recently a transaction happens, the more important the report is to those users involved.

2) *Direction:* The relation of an interaction. An edge pointed from node A to node B, which indicates that A shares some information with a set of users including B. The direction indicates that the shared information is more important to the sender than to receivers.

3) *Textual Content:* Each transaction is associated with some specific textual content, so the content of an edge means that someone thinks such content is important or related to some group of users.

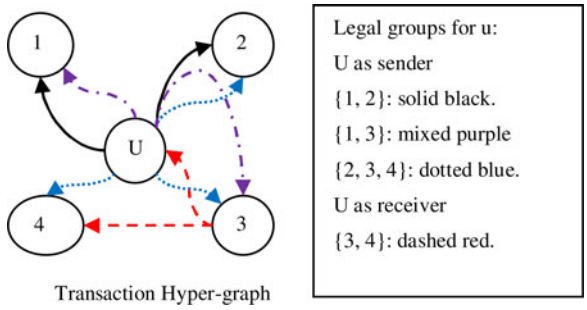


Fig. 4. Transactional user groups.

In practice, a personalized user recommendation requires the algorithm to identify potential users who have frequent and active interactions with the sender and are also interested in specific topics. In completion of two recommendation tasks, we extend both [24] and [25] by taking the direction, timeliness and textual content of the interaction into consideration to generate: 1) a suggested user list for specific report and 2) a suggested user list for specified seeds (users).

b) User groups: There could be multiple transactions associated with a specified user and each transaction involves a group of users (see Fig. 4).

Even though transactions may include the same sender and receivers, they are treated as unique in the transactional hypergraph since they are associated with unique timestamps. Despite the textual content of each transaction, the contribution of each group to current user's seeds can be evaluated by interaction rank proposed in [25].

c) User profile: To build the user profile, we consider textual content in all transactions related to the user. Carvalho [24] introduced a centroid vector-based representation, which aggregates all related documents to build a user profile. In our method, we consider transaction directions and assign document sending weight $\mathcal{W}_s$ or receiving weight $\mathcal{W}_r$ respectively. The values of $\mathcal{W}_s$ and $\mathcal{W}_r$ are manually decided based on different scenarios. For example, if the relevance of an email to the sender is higher than to the receiver (for example, junk mail or ads), then we can assign a much larger value to $\mathcal{W}_s$ than to$\mathcal{W}_r$. We use TF–IDF transformation to represent textual content as a vector. Therefore, the user profile can be represented as

$$\text{profile}\,(u) = \mathcal{W}_s \cdot \sum_{d \in S(u)} \text{tfidf}\,(d) + \mathcal{W}_r \cdot \sum_{d \in R(u)} \text{tfidf}\,(d) \tag{18}$$

where $\text{tfidf}\,(d)$ is defined as

$$\text{tfidf}\,(d)_i = \text{TFIDF}\,(d)_i^t \tag{19}$$

where $t = \frac{\text{time(now)} - \text{time}(n)}{\lambda}$ indicates an over-time exponential decay of each document's contribution. $S\,(u)\,, R\,(u)$ are sets of documents sent and received by u, respectively. Therefore, user u's preference to report d can be generated by computing the cosine similarity between the user's profile and the TF–IDF vector of d

$$\text{preference}\,(u, d) = \cos\left(\text{profile}\,(u)\,, ts_{\text{tfidf}}\,(d)\right). \tag{20}$$

Input: u, the user; d, the report, and $\mathcal{S}$, the seeds
Output: $\mathcal{R}$, recommended user list
1. $\mathcal{G} \leftarrow$ *GetTransactionalGroups(u)*
2. $\mathcal{R} \leftarrow \emptyset$
3. for each group $g \in \mathcal{G}$
4. for each user $c \in$ g, $c \notin \mathcal{S}$
5. if $c \notin \mathcal{R}$
6. $\mathcal{R}[c] \leftarrow 0$
7. $\mathcal{R}[c] \leftarrow \mathcal{R}[c]$ + *GroupScore*(c, $\mathcal{S}$, g, d)
or $[c] \leftarrow \mathcal{R}[c]$ + *CommunityScore*(c, $\mathcal{S}$, g)

Fig. 5. Suggesting user routine.

Practically, the user profile is stored separately and will not be updated in each calculation. Typically, it will be updated every few days or when new events are announced.

d) User group suggestion algorithm: We extended the friend-finding algorithm proposed in [25] to generate a list of user recommendations by aggregating the groups' contribution to a user and considering the relevance between users and reports. Our algorithm is described in Fig. 5. The score of each user in the list represents the interaction preference with respect to the given user and report.

e) Group contribution: From the algorithm described in Fig. 5, the interaction preference of a user is the aggregated value of the contribution that each transaction made to the user. There are two types of contribution measurements with respect to different tasks. We use group score and community score to represent contributions for report sharing and community user recommendation, respectively.

f) Group score: The group contribution GC described later represents the contribution that a user group contributes to a user. There are two situations considered: 1) suggesting users related to a document based on the preference (similarity) between the document and a user; and 2) suggesting a user group based on the similarity between users. We defined GC as an aggregated score of users' preferences to a specific document considering the direction and timeliness of each interaction.

For the first situation, we use similarities between each user in a group and report d:

$$\mathcal{GC}(d,g) = \mathcal{W}_s \cdot \sum_{i \in O(u,g)} s(i,d)^t + \mathcal{W}_r \cdot \sum_{i \in I(u,g)} s(i,d)^t \tag{21}$$

where $s(i,d) = \sum_{u \in i} \text{preference}(u,d)$.

For the second situation, we simply modified the $\mathcal{GC}(d,g)$ as $\mathcal{GC}(c,g)$ and $s(i,d)$ as

$$s(i,c) = \sum_{u \in i} \cos(\text{profile}(u), \text{profile}(c)) \tag{22}$$

to calculate the similarity without document information.

In both situations, $O(u,g)$ and $I(u,g)$ are sets of sending and receiving interactions/transactions, respectively, in which user u was involved.

g) Recommend users with report: To recommend a report to a group of users, one should consider historic recommendation transactions and the report's textual content. The score that a transaction contributes to a user is the aggregation of preferences of a group of users to the given report

$$\text{GroupScore}(c,\mathcal{S},g,d) = \begin{cases} \mathcal{GC}(d,g), & \text{if } S \cap g \neq \emptyset; \\ 0, & \text{otherwise.} \end{cases} \tag{23}$$

h) Recommend users for communities: Recommending users to form communities involves historic transactions without textual information. The score that a transaction contributes to a user is the aggregation of similarities between the user and users in the group

$$\text{CommunityScore}(c,\mathcal{S},g) = \begin{cases} \mathcal{GC}(c,g), & \text{if } S \cap g \neq \emptyset; \\ 0, & \text{otherwise.} \end{cases} \tag{24}$$

A user can arbitrarily choose target users at runtime. Starting from those chosen users as seeds, our recommendation components can dynamically generate more users related to the given textual content and list of users with high concurrence.

III. Case Study

A. Business Continuity Information Network

The BCiN (see Fig. 6) is a web-based prototype implementation of a Business Continuity Information Network that is able to link participating companies into a community network, provide businesses with effective and timely disaster recovery information, and facilitate collaboration and information exchange with other businesses and government agencies. The system allows company users to submit reports related to their own business, and government users to make announcements on issues impacting the public. To collect more information during the disaster, BCiN can monitor the news published on the websites and takes the news as its input. Like traditional information systems, these reports and news, and the status information of entities they contain can be retrieved and accessed by queries. For example, reports can be viewed according to alert categories or geo-locations, and resources can be viewed according to status or usages. Furthermore, BCiN not only displays user-submitted information but also conducts necessary and meaningful data processing work. BCiN makes recommendations based on the current focus and dynamically adapts based on users' interests. BCiN summarizes reports and news to provide users with brief and content-oriented stories, which prevent users from being troubled when searching through large amounts of information. By introducing the concept of community, BCIN offers users a hierarchical view of important reports or events around them.

Four main information processing and representation components are implemented in BCiN: information extraction (see Section II.A), report summarization (see Section II.B), dynamic dashboard (see Section II.C.1), and dynamic community generation (see Section II.D.1). These four different components are tightly integrated to provide a cohesive set of services and constitute a holistic effort on developing a data-driven solution for disaster management and recovery.

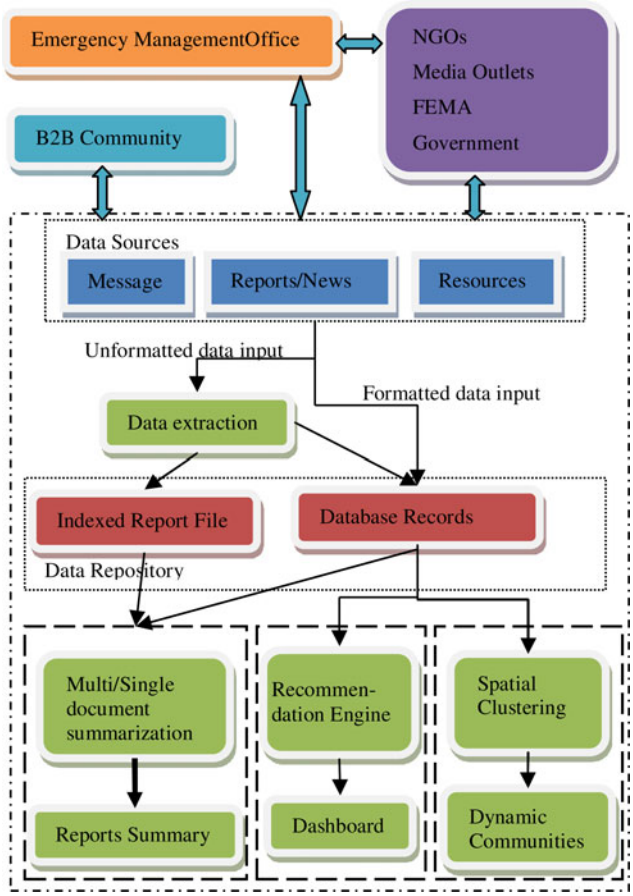


Fig. 6. BCIN system architecture.

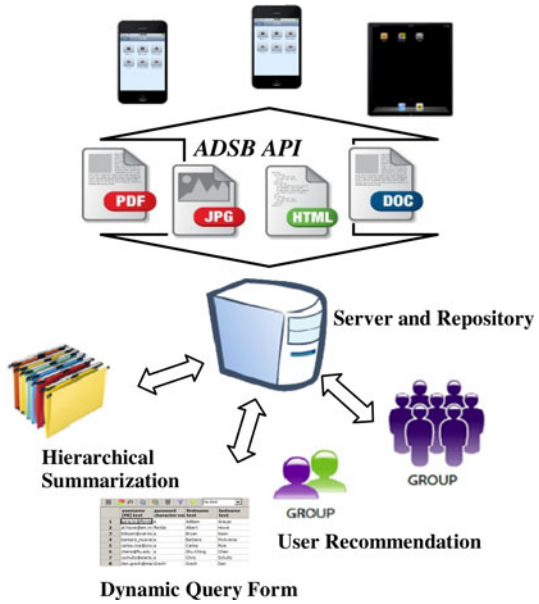


Fig. 7. ADSB system architecture.

B. All-Hazard Disaster Situation Browser

Professionals who have an operational responsibility in disaster situations are relying on mobile phones to maintain communications, update status, and share situational information. Consumers, too, are finding mobile devices convenient for sharing information about themselves and what is going on in their lives. By using a mobile platform, we can build native applications that utilize onboard sensors, rich media, and simplified user interfaces to engage users in a way that they feel is most comfortable for sharing such information in a disaster situation.

ADSB is an *All-Hazard Disaster Situation Browser (ADSB)* system that runs on Apple's mobile operating system (iOS), and iPhone and iPad mobile devices. Fig. 7 illustrates the system architecture, and Fig. 8 illustrates the system screenshot. Four major components are implemented in ADSB: information extraction (see Section II.A), hierarchical summarization (see Section II.B), dynamic query form (see Section II.C.2), and user recommendation (see Section II.D.2). A video demonstration is available at http://users.cis.fiu.edu/∼taoli/ADSB-Demo/demo.htm.

IV. System Evaluation

The data sources used in our project can be broadly divided into two categories based on temporal characteristics: static data sources and dynamic data sources. Static data sources include historical data from Miami-Dade EOC. Dynamic data sources include: 1) situation reports from Miami-Dade EOC and participating companies illustrating the current status of threat, ongoing operations, and goals/objectives for preparation and recovery efforts; 2) open/closure status about roadways/highways/bridges and other infrastructure such as fuel, power, transportation, emergency services (fire stations, police stations), schools, and hospitals; 3) reports crawled from FEMA [28] web site with information about 20 major disasters since 2000; and 4) tweets posted in August 2010 by using Twitter API [27] from dozens of active accounts.

Evaluation is conducted on two levels: algorithm evaluation and system evaluation. To evaluate the algorithms, we use standard performance metrics and compared our algorithms with existing work when applicable. Using report summarization as an example, we conducted experiments on a dataset of press releases collected from Miami-Dade EOC and Homeland Security during Hurricane Wilma from October 19, 2005 to November 4, 2005. The dataset contains 1700 documents in total, concerning all the related events before Hurricane Wilma came, during Hurricane Wilma, and after Hurricane Wilma passed [42]. The documents report various types of information such as the movement of Hurricane Wilma, the location of evacuation zones, and the cancellation of social activities. In order to evaluate the summarization performance, human generated summaries are used as references. The summarization results are evaluated by ROUGE [33].

Table VI shows the experimental results and demonstrates the efficiency of using AP to generate hierarchical document summarization (Centroid means picking the cluster centroid as the representative sentence).

Our system evaluation process consists of presenting the system to emergency managers, business continuity professionals, and other stakeholders for feedback and performing community exercises. The exercises involve a real-time simulation of

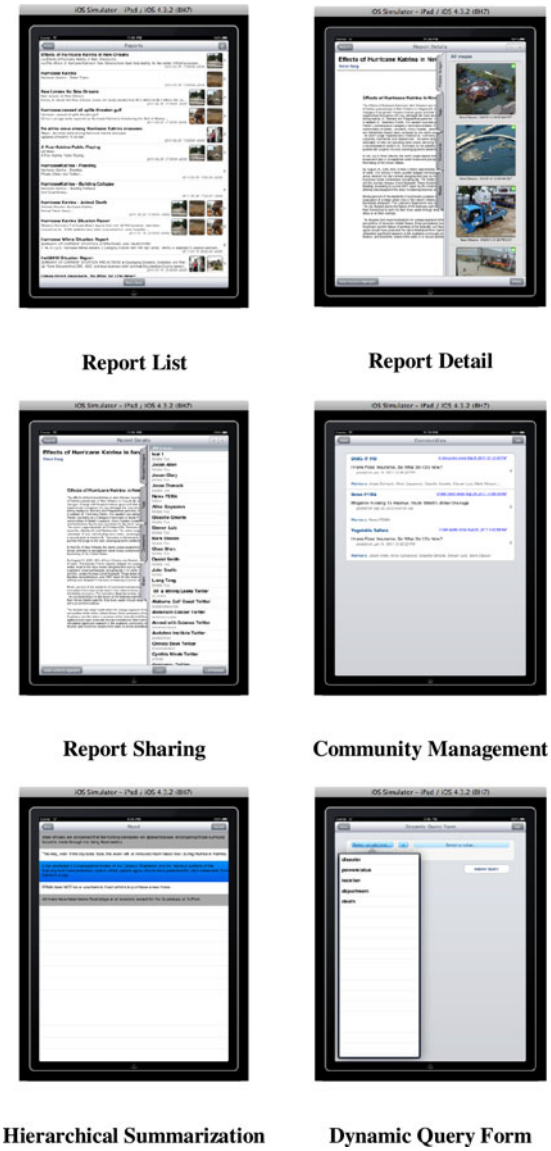


Fig. 8. ADSB screen shots of important components. *iPhone implementation has the same style with iPad but without rich visual abilities, such as the split view.

TABLE VI
SUMMARIZATION RESULTS COMPARISON

Measure		Centroid	Affinity Propagation
ROUGE-1	Recall	0.3409	**0.3788**
	Precision	0.1991	**0.3311**
	F-Score	0.2514	**0.3534**
ROUGE-2	Recall	0.0916	**0.1069**
	Precision	0.0533	**0.0933**
	F-Score	0.0674	**0.0996**
ROUGE-SU4	Recall	0.1121	**0.1173**
	Precision	0.0649	**0.1023**
	F-Score	0.0822	**0.1093**

TABLE VII
EVALUATION EXERCISES

Date	Description of the Exercise
Jun. 01 2009	In Florida Dept. of Emergency Management's Statewide Hurricane Exercises, BCiN was utilized in a scenario where Miami-Dade County Emergency Management Business Recovery Desk vafacilitated the logistics to deploy portable ATMs at Shelters and PODs in Miami-Dade County.
Jun. 29 2009	In Miami-Dade UASI exercise, BCiN supported communicating and collaborating with several companies that participated in the event as observers.
Aug. 20 2009	In a full scale company BCiN training, about 30 companies were given injects to provide information to resolve different information requests.
May 10 2010	In Miami-Dade Dept. of Emergency Management's Statewide Hurricane Exercise, our systems were responsible for disseminating and responding to injects during the course of the exercise for both government and company users.
Jul. 29 2010	In Miami-Dade company exercises, over 50 company attendees used our systems for a training exercise.
May 12 2011	In the county of West Palm Beach exercise, we demonstrated the system to WPB Dept. of Emergency Management and companies.

a disaster event integrated into an existing readiness exercise conducted each year. This evaluation exposes information at different time intervals and asks the community to resolve different scenarios by using the tool. The evaluation is a form of a "table-top" exercise in which injected information provides details about the current disaster situation and specifies potential goals and courses of action. Participant use the system to gather information to assess the situation and provide details about the actions they will take. We gather information about what information they found to derive their conclusions (or lack thereof). This information allows us to better understand how those techniques improve the information effectiveness.

Table VII describes the exercises. In a regional disaster such as a hurricane, business continuity professionals are under extreme pressure to execute their continuity of operation plans because many of the usual sources of information and services about the community and supply chain are completely disconnected, sporadic, redundant, and many times lack actionable value. The system focuses user input and collaboration around actionable information that both public and private sector can use.

To validate the usability and performance of our system, the participants and the EOC personnel at Miami-Dade participated in the questionnaire session after the exercise. A set of ten questions was designed to evaluate our system where nine of them are multiple choice questions with a five-level scale (strongly agree, agree, not sure, disagree, and strongly disagree) and the last one is an open-ended question. Some of the multiple choice questions are: Are you able to identify related reports that you are interested in? Are you able to identify the correct modules for your tasks? Are you able to switch between different modules? Are the system generated summaries useful? The open-ended question is about feedback and suggestions from the users. On average, about four EOC personnel and 30 participants attended each exercise. The evaluation demonstrated that most of participants are satisfied with the performance of the tools. Specifically, seven out of nine multiple choice questions received "strongly agree" or "agree" from over 90% of the participants, implying a high level of satisfaction with our system.

The feedback from our users is positive and suggests that our system can be used not only to share the valuable actionable information but to pursue more complex tasks like business planning and decision making. There are also many collaborative missions that can be undertaken on our system, which allows public and private sector entities to leverage their local capacity to serve the recovery of the community. We summarized the feedback as follows.

1) *Positive feedback:* a) the system is easy to use; b) related reports are well organized based on personalized user groups; and c) reports summarization is representative and interesting.
2) *Some suggestions:* a) related multimedia information, including images and video, could be shown during navigation; b) report summaries could be organized based on some points of interests.

V. Conclusion

We identified four key design challenges to support multiparty coordination during disaster situations. We proposed a unified framework that systematically integrates the different techniques that are developed in our previous work [5], [29]. Such a framework can be utilized when dealing with different systems or applications separately (e.g., BCiN and ADSB), and they are essentially collaborative platforms for preparedness and recovery that helps disaster impacted communities to better understand what the current disaster situation is and how the community is recovering. The system evaluation results demonstrate the effectiveness and efficiency of our proposed approaches.

During the system implementation and assessment process, the users provided suggestions, limitations and possible enhancements. Our future efforts will be focusing on the following tasks: developing efficient tools to automatically crawl related information from public resources including news portals, blogs, and social Medias; capturing the current user's interests and construct appropriate query form; and understanding users' intends to provide them with actionable answers to their information inquiries.

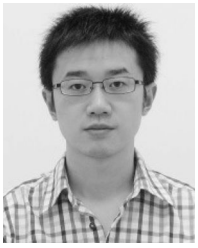
Acknowledgment

The authors would like to thank J. Domack, M. Oleson, and J. Allen for their work in the system development and testing. The initial work has been recognized by FEMA (Federal Emergency Management Agency) Private Sector Office as a model in assistance of Public-Private Partnerships [2].

References

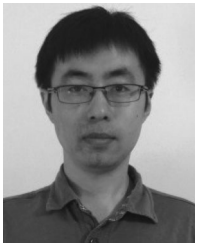
[1] H. Muson, "Preparing for the worst: A guide to business continuity planning for mid-markets," Executive Action Series, The Conference Board, Rep. A-0179-06-EA, Feb. 2006.
[2] FEMA public Private Partnership Models. [Online]. Available: http://www.fema.gov/pri-vatesector/ppp_models.shtmunder Miami-Dade County.
[3] K. Saleem, S. Luis, Y. Deng, S.-C. Chen, V. Hristidis, and T. Li, "Towards a business continuity information network for rapid disaster recovery," in *Proc. Int. Digit. Gov. Res. Conf.*, 2008, pp. 107–116.
[4] V. Hristidis, S. Chen, T. Li, S. Luis, and Y. Deng, "Survey of data management and analysis in disaster situations," *J. Syst. Softw.*, vol. 83, pp. 1701–1714, 2010.
[5] L. Zheng, C. Shen, L. Tang, T. Li, S. Luis, S. Chen, and V. Hristidis, "Using data mining techniques to address critical information exchange needs in disaster affected public-private networks," in *Proc.16th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2010, vol. 10, pp. 125–134.
[6] J. Lafferty, A. McCallum, and F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proc. 18th Int. Conf. Mach. Learning*, 2001, pp. 282–289.
[7] F. Sha and F. Pereira, "Shallow parsing with conditional random fields," in *Proc. HLT-NAACL*, 2003, pp. 131–141.
[8] C. W. Hsu and C. J. Lin, "A comparison of methods for multiclass support vector machines," *IEEE Trans. Neural Netw.*, vol. 13, no. 2, pp. 415–425, Mar. 2002.
[9] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 21, pp. 972–976, 2007.
[10] G. Erkan and D. R. Radev, "Lexpagerank: Prestige in multi-document text summarization," in *Proc. Empirical Methods Natural Language Process.*, 2004, pp. 365–371.
[11] G. Salton and M. J. McGill, *Introduction to Modern Information Retrieval*. New York, NY, USA: McGraw-Hill, 1983.
[12] P. N. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining*. Reading, MA, USA: Addison-Wesley, 2005.
[13] J. Han and M. Kamber, *Data Mining Concepts and Techniques*, 2nd ed. San Mateo, CA, USA: Morgan Kaufmann.
[14] M. Jayapandian and H. V. Jagadish, "Automated creation of a forms-based database query interface," in *Proc. VLDB*, 2008, pp. 695–709.
[15] M. Jayapandian and H. V. Jagadish, "Expressive query specification through form customization," in *Proc. 11th Int. Conf. Extending Database Technol.*, 2008, pp. 416–427.
[16] M. Jayapandian and H. V. Jagadish, "Automating the design and construction of query forms," *IEEE Trans. Knowl. Data Eng.*, vol. 21, no. 10, pp. 1389–1402, Oct. 2009.
[17] P. P. Talukdar, M. Jacob, M. S. Mehmood, K. Crammer, Z. G. Ives, F. Pereira, and S. Guha, "Learning to create data-integrating queries," in *Proc. VLDB*, 2008, pp. 785–796.
[18] H. Tong, C. Faloutsos, and J. Pan, "Fast random walk with restart and its application," in *Proc. 6th Int. Conf. Data Mining*, 2006, pp. 613–622.
[19] M. Ester, H. P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discoverying clusters in large databases with noise," in *Proc. Int. Conf. Knowl. Discovery Data Mining*, 1996, pp. 226–231.
[20] C. H. Lee, "Density-based clustering of spatial data in the presence of physical constraints," Master's thesis, Univ. Alberta, Edmonton, AB, Canada, Jul. 2002.
[21] M. D. Choudhury, W. A. Mason, Jake M. Hofman, and Duncan J. Watts, "Inferring relevant social networks from interpersonal communication," in *Proc. 19th Int. Conf. World Wide Web*, Apr. 26–30, 2010, pp. 301–310.
[22] I. Kahanda and J. Neville, "Using transactional information to predict link strength in online social networks," in *Proc. 3rd Int. Conf. Weblogs Social Media*, Jun. 2009, pp. 74–81.
[23] S. Yoo, Y. Yang, F. Lin, and I. Moon, "Mining social networks for personalized email prioritization," presented at the 15th ACM SIGKDD Int. Conf. Knowledge Discovery Data Mining, Paris, France, Jun. 28–Jul. 01, 2009.
[24] V. R. Carvalho and W. W. Cohen, "Ranking users for intelligent message addressing," presented the 30th Eur. Conf. Advances Information Retrieval, Glasgow, U.K., Mar. 30–Apr. 03, 2008.
[25] I. Horn, A. Leichtberg, N. Leiser, Y. Matias, and R. Merom, "Suggesting friends using the implicit social graph," in *Proc. Int. Conf. Knowl. Discovery Data Mining*, 2010, pp. 233–242.
[26] O. R. Aaiane, A. Foss, C. H. Lee, and W. Wang, "On data clustering analysis: Scalability, constraints, and validation," in *Proc. 6th Pacific-Asia Conf. Adv. Knowl. Discovery Data Mining*, 2002, pp. 28–39.
[27] Twitter API. [Online]. Available: http://apiwiki.twitter.com
[28] FEMA. [Online]. Available: http://www.fema.gov
[29] L. Zheng, C. Shen, L. Tang, T. Li, S. Luis, and S. Chen, "Applying data mining techniques to address disaster information management challenges on mobile devices," in *Proc. Int. Conf. Knowl. Discovery Data Mining*, 2011, vol. 11, pp. 283–291.
[30] D. McEntire, "The status of emergency management theory: Issues, barriers and recommendations for improved scholarship," presented at the FEMA Higher Education Conf., Emmitsburg, MO, USA, 2004.
[31] GeoVISTA. [Online]. Available: http://www.geovista.psu.edu
[32] C. X. Zhai and J. Lafferty, "A study of smoothing methods for language models applied to ad hoc information retrieval," in *Proc. SIGIR*, 2001, pp. 334–342.
[33] C. Lin, "Rouge: A package for automatic evaluation of summaries," in *Proc. Workshop ACL*, 2004, pp. 25–26.

[34] E. Brill, "Part-of-speech tagging," in *Handbook of Natural Language Processing*. Boca Raton, FL, USA: CRC Press, 2000, pp. 403–414.
[35] The Puerto Rico Disaster Decision Support Tool (DDST). [Online]. Available: http://www.udel.edu/DRC/DDST/
[36] E. J. Bass, L. A. Baumgart, B. Philips, K. Kloesel, K. Dougherty, H. Rodríguez, W. Díaz, W. Donner, J. Santos, and M. Zink, "Incorporating emergency management needs in the development of weather radar networks," *J. Emergency Manage*., vol. 7, no. 1, pp. 45–52, 2009.
[37] L. A. Baumgart, E. J. Bass, B. Philips, and K. Kloesel, "Emergency management decision-making during severe weather," *Weather Forecasting*, vol. 23, no. 6, pp. 1268–1279, 2008.
[38] C. E. League, W. Díaz, B. Philips, E. J. Bass, K. A. Kloesel, E. C. Gruntfest, and A. Gessner, "Emergency manager decision-making and tornado warning communication," *Meteorological Appl*., vol. 17, no. 2, pp. 163–172, 2010.
[39] WebEOC. [Online]. Available: Manufactured by ESi Acquisition, Inc. http://www.esi911.com/home
[40] E-Teams, by NC4. [Online]. Available: http://www.nc4.us/ETeam.php
[41] National Emergency Management Network. [Online]. Available: http://www.nemn.net/
[42] L. Li and T. Li, "An empirical study of ontology-based multi-document summarization in disaster management," *IEEE Trans. SMC: Syst*., 2013, in press.
[43] L. Tang, T. Li, Y. Jiang, and Z. Chen, "Dynamic query forms for database queries," *IEEE Trans. Knowl. Data Eng*., 2013, in press.



Li Zheng received the B.S. and M.S. degrees in computer science from Sichuan University, Chengdu, China, in 2004 and 2007, respectively. He is currently working toward the Ph.D. degree with the School of Computer Science, Florida International University, Miami, FL, USA.

His research interest includes vertical search engine, recommender system, and disaster management.



Chao Shen received the B.S. and M.S. degrees in computer science from Fudan University, Shanghai, China, in 2006 and 2009, respectively. He is currently working toward the Ph.D. degree with the School of Computer Science, Florida International University, Miami, FL, USA.

His research interest includes text mining, text summarization, and data mining of social media.

Liang Tang received the B.S. and M.S. degrees in computer science from Sichuan University in 2006 and 2009, respectively. He is currently working toward the Ph.D. degree with the School of Computer Science, Florida International University, Miami, FL, USA.

His research interest includes event mining, large scale data mining, and recommender systems.

Chunqiu Zeng received the B.S. and M.S. degrees in computer science from Sichuan University, Chengdu, China, in 2006 and 2009, respectively. He is currently working toward the Ph.D. degree with the School of Computer Science, Florida International University, Miami, FL, USA.

His research interest includes large scale data mining, event mining and text mining.

Tao Li received the Ph.D. degree in computer science in 2004 from the University of Rochester, Rochester, NY, USA.

He is currently an Associate Professor with the School of Computer Science, Florida International University, Miami, FL, USA. His research interests include data mining, machine learning and information retrieval.

Dr. Li received the USA NSF CAREER Award and multiple IBM Faculty Research Awards.

Steve Luis received the Master degree in computer science from Florida International University (FIU), Miami, FL, USA, in 1998.

He is currently the Technical Lead with FIU's Disaster Information Technologies Research Group responsible for the software architecture, requirements, and design for many of the group's core technology tools such as the Business Continuity Information System and the Mobile Disaster Situation Browser. He also conducts business development and partner outreach with more than 100 company and government agencies as part of several public/private partnerships for business recovery in South Florida.

Mr. Luis is recognized for his contribution to the resilience of South Florida communities by the Miami-Dade Emergency Management and Palm Beach County Division of Emergency Management.

Shu-Ching Chen (M'95–SM'04) received the Ph.D. degree in electrical and computer engineering in 1998, and the Master's degrees in computer science, electrical engineering, and civil engineering in 1992, 1995, and 1996, respectively, all from Purdue University, West Lafayette, IN, USA.

He is currently a Full Professor with the School of Computing and Information Sciences, Florida International University, Miami, FL, USA. His main research interests include content-based image/video retrieval, distributed multimedia database management systems, multimedia data mining, multimedia systems, and Disaster Information Management.

Dr. Chen received the 2011 ACM Distinguished Scientist Award. He received the Best Paper Award from the 2006 IEEE International Symposium on Multimedia. He is a Fellow of SIRI and was a steering Committee Member for the IEEE TRANSACTIONS ON MULTIMEDIA from 2011 to 2013.