Automating Incident Analysis: A Challenge Paper

Fergus Toolan; Computer Science Dept., UCD; Dublin; Ireland
Joe Carthy; Computer Science Dept., UCD; Dublin; Ireland
Anne Drummond; Center for Safety and Health, UCD; Dublin; Ireland
John Dunnion; Computer Science Dept., UCD; Dublin; Ireland

Abstract

Government regulations have made the recording of incidents a legal requirement in many organisations. This data is kept for no purpose as most organisations never examine the information held in incident management systems. This paper examines possibilities for the automated analysis of these incident collections, hoping to learn from what has occurred previously to prevent future incidents and accidents.

Introduction

This paper discusses the area of automated incident analysis - the extraction of knowledge from an incident management system. We discuss some possible methods to use in this field and challenge the A.I. and I.R. communities to develop novel approaches to the automatic analysis of incidents.

This paper is structured as follows. We begin with an overview of Incident Management Systems and discuss some aspects of incident reporting and incident retrieval. Following this we introduce some possible methods to automate the incident analysis process. We then examine new approaches to evaluating the quality of the algorithms. It is hoped to distance ourselves from the requirement of having a domain expert present to decide the relevance of individual incidents. Next we discuss work relevant to this paper. Finally we summarise our contributions and mention some of the directions the authors are considering investigating.

Incident Reporting and Analysis

This section examines the two main components of Incident Management Systems: Incident Reporting and Incident Analysis. The following sections discuss these in detail and describe the current state of the art techniques in these areas.

Incident Reporting: Johnson [8] states that incident reporting schemes are increasingly being seen as a means of detecting and responding to failures before they develop into major accidents. He identifies seven benefits of using these schemes [7]. We focus on two of them:

- Incidents help to find out why accidents did not occur.
- The higher frequency of incidents permits quantitative analysis

These points suggest the need for automated incident analysis as according to the second point we have a large collection of incidents available to us. In most domains the incident collection is too large to examine manually and hence there is a need to automate the task.

The first point suggests the importance of automating the incident analysis task as it can prevent future accidents. These show the motivation for our work: the analysis of large collections of incidents to prevent future accidents.

However there are some problems with Incident Reporting Schemes that must be addressed. One of the big drawbacks in incident reporting is the feeling amongst staff that the system will be used to assign blame [13]. This in turn leads to people not submitting incident reports or submitting incorrect incident reports. This is a drawback to incident analysis be it manual or automated. If the reports are incorrect the analysis can never be accurate.

Root Causal Analysis: Root Causal Analysis is a means of identifying the causes of incidents / accidents using semi-rigorous methods to achieve this goal.

Safety through Organisational Learning (SOL) is an event analysis technique based on concepts of the socio-technical systems approach (STSA) and assumptions about accident causation [18] and Reason's Theory of Causation [14].

SOL uses a systemic view of safety, identifying five systems that must interact correctly to ensure safety of the system. The five are technology, individual, organisation, working group and organisational environment.

Why-Because Analysis: WBA [10] is one commonly used method of Root Causal Analysis. Indeed its use in industry is increasing with companies such as Siemens making use of the techniques in the analysis of accidents in the German rail industry [3]. WBA can be sub-divided into two important areas: 1) the creation of the WB Graph and 2) the verification of this graph.

The WB Graph allows us to identify the root causes of an incident / accident. It is in the area of WBA that Accident Investigators are most interested as it helps them to understand the causes of an accident. The verification step while much more complex gives a rigorous `proof' of the correctness of the WBG. It can be used to identify missing events / states in the graph. The ability to verify the graphs is the most striking difference between WBA and other Root Causal Analysis Techniques.

According to [9] the method used to generate the WBG is as follows

- **List:** List all events and states
- **Determine Causal Factors:** Use the Causal Factor test to determine the causal factors of an accident.

The second step in the technique is verification. This consists of a formal proof, the purpose of which is to ensure that

- the causal relations asserted by the WBG are correct, and
- that a sufficient amount of factors have been identified to provide enough support for the results of the WBG.

While the verification step is a difficult mathematical process, the power (and confidence) it gives are extremely important. The formal nature of the technique leads to the belief that it should be possible to automate this stage of the process.

Approaches to automated analysis

The major difficulties in automated incident analysis come from the standard representation of an incident that most Incident Management Systems use. These usually rely on small amounts of field-based information such as date, time, location...etc. and on large quantities of textual description in the form of witness reports, ATC reports...etc. The field-based information lends itself instantly to automated analysis however it is unlikely that using this information will result in any useful trends being discovered. The interesting information is often contained in the free text descriptions of the incidents.

To utilise this freetext information effectively for automated analysis we need to gain a better representation of this information. The challenge of the representation phase of automated analysis is to take the free text documents and extract the relevant information from them and store it in a format that can be used in pattern extraction algorithms.

This section is laid out as follows. Firstly we will look at the representation issues in incident reporting. This section also looks at how to translate from the human readable format that the report is in to a more machine-readable form. Following on from that we examine techniques for the automated analysis phase itself. Here we look at some established techniques such as Case Based Reasoning (CBR) and Data Mining. We also investigate new techniques in this area.

Representation: As already stated Incident Reporting schemes result in a combination of data types. We have numeric data such as the number of injuries or the number of people involved in an incident. We see temporal information in the form of dates, times and time intervals. Incident reporting schemes also result in textual information in two major formats: small text field information such as location and large textual narratives such as the description of the event. The heterogeneous nature of the information leads to difficulties in analysing the information. The techniques we discuss in the following sections are more suited to dealing with numeric fields and small textual fields. The major challenge lies in the fact that some of the most useful information is contained in the textual description fields of the report. This information is difficult to translate into a machine usable form.

Numerous techniques are possible to create a better representation of this information. We are mainly concentrating on WBA but any Root Causal Analysis technique may be used. We suggest that the WBG of the description may be the best representation to use. The graph representation is relatively straightforward to `mine' information from as will be seen later. However the transformation from textual to graphical representation may be quiet difficult to achieve.

Possibilities at this stage include a basic Information Retrieval approach using such techniques as stopword removal, stemming or n-gram extraction. However basic information retrieval loses some information vital to the incident analysis task - temporal information. We argue that it becomes impossible to accurately represent an incident without knowledge of the sequencing of the events. Take for example the process of starting a car. The driver ensures the car is in neutral, starts the engine, puts the car in first gear, releases the handbrake and drives away. A driver following these steps will successfully start the car. However, if instead the driver placed the car in gear, and tried to start the car it would stall. The driver in the second example is performing steps necessary to start a car but the sequencing means that they will never be able to do it.

This would suggest that Information Retrieval techniques need to be enhanced with something extra. Natural Language Processing provides some knowledge of sequencing of events. This sequencing information is vital as with correct sequencing we narrow the possibilities for the root causal identification task. We know that a root cause of an event must occur before the event (or the state) it lead to and hence any event with no parent in the temporal hierarchy is a possible root cause.

To give a more formal representation of this we say if *A* and *B* are events in an incident description and if *A* occurs before *B* then *B* cannot be a root cause of the incident. This does not however, mean that *A* is a root cause only that *A* may be a root cause. When we have identified the WBG for the incident we are ready to combine this with the other fields in the incident management system relating to the incident in question and gain our final, complete description of the incident in a machine-readable form.

Automated Why-Because Analysis: The automation of the graph generation phase of WBA is one of the most complex tasks in automated incident analysis. The methods used to perform WBA do not lend themselves well to automatic execution by the computer. The first step, that of listing the events / states in the incident report is a very difficult natural language processing task. However with the nature of the aviation domain the task is semi-constrained making it much easier.

An example of this is the natural language problem of synonyms. Synonyms are two words that have the same meaning. An example of this in the aviation domain are the words **plane** and **aircraft**. However the technical nature of the domain has lead to each reporting scheme standardising this type of terminology. For instance the ASRS [19] dataset uses the term ACFT to mean airplane, plane or aircraft.

This constraint and others like it will hopefully allow more structured Information Retrieval techniques to be applied to the data rather than NLP techniques. Techniques such as stopword removal, stemming and n-gram extraction in conjunction with the most basic NLP techniques may be helpful in generating the events / states that occurred in the incident.

Case Based Reasoning: Case Based Reasoning (CBR) techniques can be used both to analyse incidents and to retrieve them from the incident management system. When used for retrieval they provide a `fuzzier' matching criterion than standard exact-match database queries [4]. In records as large as NASA's Aviation Safety Reporting System (ASRS) this is a vital facility as these records are too large to find exact matches. It can also be used for such things as judging the similarity between incident fields. For instance an incident involving a Boeing 737 would be more similar to one involving a Boeing 747 than a Cessna.

CBR techniques have been widely used to support a number of decision making tasks [7] such as faultfinding in the aviation domain. The decision necessary in incident analysis is the similarity of a new incident to others that occurred previously. These systems sometimes use a method known as Conversational CBR where the system has a set of questions it asks the user on encountering a new case. For instance in the technical support domain in the IT industry a question may be ``Does the monitor flicker?'' Based on the answer the categorisation of the new problem gets one step closer to being correct.

NaCoDae (Navy Conversational Decision Aids Environment) uses Conversational CBR to discover incidents similar to a users query terms [1]. It uses a free text case representation which includes the appropriate solution to the problem if available. NaCoDae gradually refines the users query through use of conversational techniques and hence overcomes inaccuracies in the users query.

Data Mining: Data Mining techniques are used to discover patterns in large Database collections [2]. In the domain of Incident Analysis they can be used to generate some basic patterns that are not necessarily obvious to the human user. In [16] we look at this area in more detail. This section gives an overview of Association Rule mining a very common technique in Data Mining. These focus on field-based information available in the dataset as oppossed to text-based information.

An Association Rule is a rule which implies certain association relationships amongst a set of objects in a database [2]. For instance, association rules could develop a set of symptoms associated with a disease or a set of items that commonly co-occur in a shopping basket.

Let L be a set of Literals (or items). An association rule is of the form $X \rightarrow Y$, where X, $Y \subseteq L$. The meaning of $X \rightarrow Y$ is that transactions that contain X tend to contain Y.

An association rule has two numeric terms associated with it namely its confidence and its support. An example rule is that ``30% of transactions that contain beer also contain diapers: 2% of all transactions contain both of these items". We can define confidence and support in terms of this rule. The 30% value is the confidence. It is the number of transactions which contain X and also contain Y. The support value of 2% measures the percentage of occurrences of *both* X and Y in the set L. The problem is therefore to find all association rules that satisfy user-specified minimum support (S_{MIN}) and confidence (C_{MIN}).

Countless algorithms have been proposed for association rule mining. The best known is the *Apriori* algorithm which divides the problem into two separate parts

- Find combinations of items that have a transaction support above minimum support. These are frequent itemsets.
- Use frequent itemsets to generate the desired results. To do this assume $X \cup Y$ and $X \cup Y \cup Z$ are frequent itemsets then we can see if $(X \cup Y) \rightarrow Z$ holds by computing r, the ratio of $sup(X \cup Y \cup Z)$ to $sup(X \cup Y)$.
- If $r \geq C_{MIN}$ then $(X \cup Y) \rightarrow Z$ is a valid rule.

Many other algorithms exist for mining association rules. These include modifications to Breadth-First Search and Depth-First Search, and partition algorithms [5]. However, the more common solutions involve *Apriori* or new variations on the algorithm.

Classification Based Techniques: Classification based techniques are a standard Machine Learning technique that are used to decide how an item should be classified based on rules learned from a pre-classified set of items. Many forms of algorithms exist in this area. A recent application of basic classification techniques appeared in [6] using the horse racing domain.

Classification examines a set of data and generates a set of classification rules by which we can classify future data. This is very much in common with statistics and machine learning. In classification one develops a description or model for each class in a database based on the features present in a set of class-labeled training data [15].

Various methods exist for mining classification rules [12]. The simplest forms are statistical algorithms such as linear models found in such packages as SAS or SPSS however, these don't scale very well. Another method is that of Neural Networks which try to copy the pattern matching ability of the human brain. Yet another commonly used technique is that of the nearest neighbour algorithm. This classifies each record in the dataset based on a combination of the classes of the *k* records most similar to it in a historical dataset. Another technique is rule induction which is the extraction of *if - then* rules from data based on statistical significance.

Graph Matching Algorithms: The techniques we have looked at so far for the analysis rely on the field-based information in the dataset and the new field-based representation of the narrative descriptions. However if root causal analysis is automated our final product contains the WBG (Why-Because Graph). If the automation of WBA is successful this will give us a new data format to try to analyse. The most likely candidates to be used in analysing this data is that of Graph Matching algorithms. Given that we are comparing two events Ev_1 and Ev_2 where $G_1 = (V_1, E_1)$ is the WBG for Ev_1 and $G_2 = (V_2, E_2)$ is the WBG for Ev_2. Let us define some notation for this: if G_1 is a subgraph of G_2 i.e. if $V_1 \subseteq V_2$ and $E_1 \subseteq E_2$ we write this as $G_1 \subset_G G_2$. If we wish to say that G_1 is a similar-subgraph of G_2 we write it as $G_1 \subset^{\approx}_{G} G_2$.

We are looking for a graph G_R where

$$G_R \subset_G G_1 \text{ and } G_R \subset_G G_2$$

or

$$G_R \subset^{\approx}_{G} G_1 \text{ and } G_R \subset^{\approx}_{G} G_2$$

The notion of a subgraph is a standard graph theoretic term. The above equations are saying that we are looking for a graph G_R which is a subgraph of both G_1 and G_2 or a similar-subgraph of G_1 and G_2. The notion of a similar-subgraph will be explained later.

Common Subgraphs Algorithm: The algorithm for extracting the common subgraphs of two graphs is shown in Figure 1. It begins by finding X_V and X_E the intersections of the Vertex set and the Edge set respectively. Then for every element of X_V it checks for members of X_E which are of the form (v_i, Z) where v_i is the current element of X_V and Z is any other element of X_V. When it finds these it adds v_i and Z to V_R - the vertex set of G_R - and adds the edge (v_i, Z) to the edge set of G_R.

This algorithm can extract all common subgraphs from two graphs with a complexity of $O(nm)$ where n is the cardinality of the intersection of the vertex sets, X_V, and m is the cardinality of the intersection of the edge sets, X_E.

Similar Subgraph's: In the domain of incident analysis the WBG is one of the best methods of visualising the sequence of events and the causal factors. When comparing WBG's we argue that the common subgraph is too restrictive a method to use. It is feasible to imagine a situation where an event, E_1 has a simple subgraph such as $A \rightarrow B \rightarrow C$ where $A \rightarrow B$ means A was a causal factor of B and event E_2 has as a subgraph $A \rightarrow C$. It is clear that these are not equivalent subgraphs however we argue that they are related.

Given 2 Graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$
Problem: Find G_R - the common subgraph
of $G_1 \wedge G_2$

Calculate X_V : $X_V = V_1 \cap V_2$
Calculate X_E : $X_E = E_1 \cap E_2$
$G_R := \emptyset$

$\forall\ v_i : v_i \in X_V$

$\forall\ e_j : e_j \in X_E \wedge e_j = (v_i, Z)$
$AddVertex(G_R, v_i)$
$AddVertex(G_R, Z)$
$AddEdge(G_R, v_i, Z)$
END
END

Figure 1 – Common Subgraph Algorithm

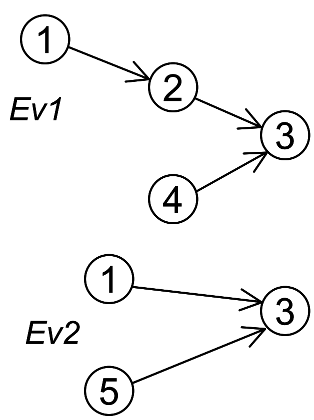

Figure 2 – WBG's for Ev_1 and Ev_2

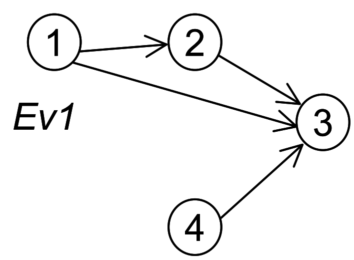

Figure 3 – Path Connected Ev_1

In both cases the event A lead either directly or indirectly to the event C. We argue there is a commonality here that the strict notion of common subgraphs would not allow us to exploit. We introduce the notion of similar subgraphs. Figure 2 shows two WBG's where states are represented by numerals.

Due to the nature of Causal graphs we can simplify the similar-subgraph detection problem by creating a ``path-connected" graph. The fact that WBG's are directed graphs allow us to connect the graphs along paths so every node in a maximal length path is connected in the direction of traversal. In Figure 2 Ev_2 has 2 maximal length paths of length 2, $1 \rightarrow 3$ and $5 \rightarrow 3$, while Ev_1 has 2 maximal length paths, one of length 2, $4 \rightarrow 3$, and one of length 3, $1 \rightarrow 2 \rightarrow 3$.

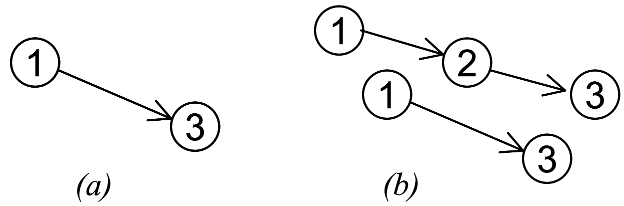

Figure 2 – (a) Common Subgraph (b) Similar Subgraphs

All maximal length paths of length 2 are by definition connected so in the above example the only unconnected maximal length path is $1 \rightarrow 2 \rightarrow 3$. Figure 3 shows the connected version of Ev_1. Creating the edge $1 \rightarrow 3$ is the only necessary step in creating the connected path. We can now use our common subgraph method to find common subgraphs and then editing them to get similar subgraphs. Figure 2(a) shows the common subgraph between Ev_2 and the connected Ev_1 while Figure 2(b) shows the similar subgraphs.

To generate the similar subgraphs from the common subgraph in Figure 2(a) we need to compare the common subgraph to the Ev_1 and Ev_2 graphs in Figure 2 and calculate any edges used in the common subgraph that were inserted at the connection phase. In this case we used $E(1,3)$ from Ev_1. We need to find the original path: this was $1 \rightarrow 2 \rightarrow 3$, and hence we have the simlar subgraphs shown in Figure 2(b).

Evaluation

Methods of evaluation can be quiet time consuming in these fields. A standard method of evaluation is to run the algorithms over a pre-classified set of examples and observe the level of accurracy it obtains. This is extremely difficult to achieve as it requires considerable person-hours from a domain expert to decide which incidents are related to each other.

However we have no valid method to avoid this method of evaluation. We can not evaluate the reliability of our algorithms without comparing them to a pre-classified set of incidents. This classification must be performed by a domain expert. It is possible to speed up the process by getting a domain expert to classify a small subset of the dataset and using a clustering algorithm to group other unclassified instances into these clusters.

Related Work

There is much work in the fields of Root Causal Analysis, CBR, Data Mining and Classification. However in terms of the application of CBR, Data Mining and Classification related to the Incident / Accident Analysis domains little has been done.

There is much work both in academia and industry on Root Causal Analysis. Numerous techniques are being used on a day-to-day basis. Such techniques include SOL [18], STAMP [11], and WBA [9] [10]. These techniques have been adopted (and altered) by industry and have been put to work in domains such as aviation [10] in the case of WBA, the nuclear power industry [18] in Germany uses SOL and the German Rail Industry where Siemens use a simplified version of WBA [3]. Work continues at pace both in refining the techniques used for Root Causal Analysis and the actual application of these techniques to discover the causes of accidents around the globe.

Data Mining has been applied to many domains from Horse Racing [6] to market basket data [2]. Both standard association rules [2] and more specialised techniques such as sequential pattern mining [15] are applicable in the incident analysis domain. Association rules are capable of generating commonly co-occurring items in a dataset while sequential patterns can predict common patterns of a sequential nature.

We are not altogether certain of the benefits of using association rule mining. Hobson-Shaw [6] found that in a dataset with many fields describing a single record that association rules were too general. For instance it would be feasible that the rule ``if the aircraft has an engine then it will crash" could be generated. Such a rule (while true based on the data examined i.e. every plane involved in an incident did have an engine) is too simplistic to be used in the real world. This leads us on to classification. Classification allowed [6] to discover more informative results.

Classification techniques such as Nearest Neighbour techniques and Decision Tree Learning Algorithms have been used regularly to classify items in a dataset into various categories. A new and interesting application of these techniques appeared recently in the form of classifications of winners from horse racing results in England [6].

As regards Case Based Reasoning, Cassidy et al [4] use CBR in a retrieval system for similar incidents in an incident management system. This can however, be viewed as a basic form of Incident Analysis in that the mere act of defining a similarity measure between two incidents is a method of analysing them. In this area the CBR methods outperformed the standard exact match methods of retrieval.

Conclusions

This paper has presented the challenge of automated analysis of incident report archives. Our work continues to focus on the area of automating Incident Analysis techniques such as WBA and the application of Data Mining techniques in the domain. In [16] we give some preliminary results from the application of Data Mining techniques to the domain. These by themselves are not extremely effective however we envisage a situation where these techniques used in conjunction with the methods of automating such techniques as WBA might prove to be extremely reliable.

References

[1] Aha, D. W., Breslow, L. A. & Maney, T. ``Conversational Case-Based Reasoning" in D.W. Aha & J.J. Daniels (Eds.) Case-Based Reasoning Integrations: Papers from the 1998 Workshop (Technical Report WS-98-15), 1998.

[2] Agrawal, R. & Srikant, R. ``Fast Algorithms for Mining Association Rules" in Proceedings of the 20th VLDB Conference, Santiago, Chile, 1994.

[3] Braband, J. & Brehmke, B. ``Application of Why-Because Graphs to Railway Near Misses" In Proceedings of Investigation and Reporting of Incidents and Accidents, Glasgow, Scotland, 2002.

[4] Cassidy, D., Carthy, J., Cullen, C., Hillary, P., McAdam, S., McGinty, L. & Sheppard, J. ``The Design and Implementation of an Incident Report Retrieval System" In Proceedings of the 1st STarting AI Researchers Symposium, (STAIRS 2002) held in conjuction with ECAI 2002, Lyon, France, 2002.

[5] Hipp, J., Guntzer, U. & Nakhaeizadeh, G. ``Algorithms for Association Rule Mining - A General Survey and Comparison." SIGKDD Explorations, 2000.

[6] Hobson-Shaw, L. ``V. Wright - The Backer's Friend." Final Year Project Report submitted to the Computer Science Department, UCD, 2003.

[7] Johnson, C. ``Using Case-Based Reasoning to Support the Indexing and Retrieval of Incident Reports" in Proceedings of European Safety and Reliability Conference (ESREL), Rotterdam, 2000.

[8] Johnson, C. ``Software Tools to Support Incident Reporting in Safety-Critical Systems" in Journal of Safety Science, Elsevier, Vol 40, Number 9, 2002.

[9] Ladkin, P. B. ``A Quick Introduction Why-Because Analysis" Available at http://www.rvs.uni-bielefeld.de/research/WBA/, 1999.

[10] Ladkin, P. B. ``Causal Analysis of Aircraft Accidents" Invited Paper in Computer Safety, Reliability and Security, Proceedings of the 19th International Conference, SAFECOMP 2000, Lecture Notes in Computer Science No. 1943, Springer-Verlag, 2000.

[11] Levenson, N. ``A New Accident Model for Engineering Safer Systems" in Proceedings of the MIT Engineering Systems Division Internal Symposium, May, 2002.

[12] Mitchell, T. M. ``Machine Learning", McGraw-Hill, 1997

[13] O'Kane, M. ``Data Driven WebSite for Incident Management" Final Year Project Report submitted to the Computer Science Department, UCD, 2003.

[14] Reason, J. ``Human Error" Cambridge University Press, 1991.

[15] Toolan, F. & Kushmerick, N. ``Mining Web Logs for Personalized Site Maps" in proceedings of the Workshop on Mining for Enhanced Web Search, International Conference on Web Information Systems Engineering (Singapore), 2002.

[16] Toolan, F. & Carthy, J. ``Data Mining Techniques applied to the Incident Analysis Domain" IIRG Technical Report - TR-IIRG-03-01, 2003.

[17] Witten, I. H. & Eibe, F. ``Data Mining: Practical machine learning tools with Java implementations" Morgan Kaufmann, San Francisco, 2000.

[18] ``Root Causal Analysis Literature Review" Prepared by WS Atkins Consultants Limited for the Health and Safety Executive, 2001.

[19] Aviation Safety Reporting System (ASRS) http://asrs.arc.nasa.gov/

Biographies

Mr. Fergus Toolan, Intelligent Information Retrieval Group, Dept. of Computer Science, UCD, Dublin, Ireland.; telephone +353.1.716.2908; fax +353.1.269.7262; e-mail - fergus.toolan@ucd.ie.

Fergus Toolan is a PhD student in the Computer Science department in University College Dublin, Ireland. His research interests include Automated Incident / Accident Analysis, Data Mining and Information Retrieval.

Dr. Joe Carthy, Intelligent Information Retrieval Group, Dept. of Computer Science, UCD, Dublin, Ireland.; telephone +353.1.716.2481; fax +353.1.269.7262; e-mail - joe.carthy@ucd.ie.

Joe Carthy has been a lecturer in Computer Science at UCD since 1984. He has been actively involved in Information Retrieval research. His research interests include Topic Detection and Tracking, Incident Analysis and Retrieval and Lexical Chaining.

Ms. Anne Drummond, Centre for Safety and Health at Work, UCD, Dublin, Ireland.;
e-mail - anne.drummond@ucd.ie

Anne Drummond, MSc (Occupational Health), DipSHWW, RGN, RM, has worked in the UCD Centre for Safety and Health at Work since 1995 following a career in the healthcare sector. She is a College Lecturer, and Joint Academic Director of the Centre for Safety and Health at Work, which is an academic centre within the Faculty of Science.

Mr. John Dunnion, Intelligent Information Retrieval Group, Dept. of Computer Science, UCD, Dublin, Ireland.; telephone +353.1.716.2474; fax +353.1.269.7262; e-mail - john.dunnion@ucd.ie.

John Dunnion has been working in the Department of Computer Science in UCD since 1985 and has been on the academic staff since 1991. His principal research interests are in the area of intelligent text-based information retrieval.