Transactions on Machine Learning
and Data Mining
Vol. 9, No 2 (2016) 62-77
© ISSN: 1865-6781 (Journal)
ISBN: 978-3-942952-47-7
IBaI-Publishing ISSN 1864-9734

Innovations in News Media: Crisis Classification System

David Kaczynski, Lisa Gandy, and Gongzhu Hu

Central Michigan University,
Mt Pleasant, MI 48858

Abstract. Research in crisis management is a relatively new area of study, originating in the 1980s. Researchers have created several different models that separate organizational crises into discrete stages, such as pre-crisis, crisis and post-crisis. In this article we discuss a natural language based crisis detection system which classifies news articles relating to crises into the appropriate crisis stage. We use news articles from the New York Times as a source of training data, and use this data along with state of the art data mining and machine learning algorithms as the core of the system. In the future, our system may be expanded to identify and evaluate crisis management strategies, suggest crisis management strategies for the current state of a crisis, or provide stakeholders with summaries of crises in news media.

Keywords: Data Mining, Machine Learning, Crisis Management

1 Introduction

A major research area in crisis communication is how to prevent or repair damage to the reputation of organizations before, during, or after the occurrence of crises. The creation of various crisis communication theories have been facilitated through case studies and by measuring the financial and/or reputation repair results of applied strategies. In this article, we explore the use of data mining techniques in identifying characteristics of news articles relating to organizations in crisis. While there are many types of crises, staged crisis models, and crisis communication strategies, we begin our research by attempting to classify organizational data as pertaining to a specific stage of a crisis. After using data mining and classification techniques we create a system which can take any news article and classify it as pre-crisis, crisis or post-crisis.

We use news media as a source of data for training in our system. The methodology that we apply can be repeated in order to add more articles to the data set in the future and retrain the model. This may increase the accuracy for classifying crises that are not yet in the training data. Our methods may also be repeated for the purpose of adding more labels in order to classify more specific characteristics of the crises or of the crises response strategies being employed. There are several contributions of this paper. Though incident management systems do exist [6], [19], [30], our incident management system is built on a major theory of crisis communication, the Coombs model [8]. This model divides crises into three stages: pre-crisis, crisis and post crisis. Our system focuses on incident identification using news articles, which are widely prevalent due to the Internet, whereas other systems use numeric data. Our system requires no data preparation, as it can classify the text of any article, whereas again other systems often require the use of numeric data which must be prepared before use. In addition, our system has been evaluated on many sets of crises, whereas other systems often focus on one type of crisis, such as crime [6], traffic accidents [20], or weather [30]. In summary, our work focuses on a natural language based real time crisis identification system which requires no data preparation by the user and has already been evaluated on several existing crises. In the background section we provide the preliminary information for the research. The section about our system walks through the proposed data mining process in detail including experimental results and the evaluation of the results. We then discussed the implementation of our system as a Web application, followed by a briefly discuss current research in crisis management and data mining. We conclude with a discussion of a proposed system based on our methodology and discuss future work.

2 Related Work

Researchers have studied crisis management for many years and quite a number of systems were developed. In this section, we discuss some previous work related to crisis management that focused on text mining.

2.1 Crisis Detection Systems

Current crisis detection systems focus on using data mining techniques to either identify or predict a type of crisis. Peng, et al [30], introduced a three stage system which ultimately uses Multicriteria Decision Making to determine the risk of an incident occurring. This system was demonstrated using agrometeorological disaster data. The system was in the conceptual stage and has not been implemented. Chen, et al. [6] proposed a general framework for crime detection. Their system was well designed and worked well for the give data set, however it differs from our work as it focuses on the detection of new crimes using numeric data, whereas our system focuses on detecting corporate crises using text data in the form of news articles.

Similar to the work in [30] and [6], Harms et al. [15] used association rule mining to find patterns in numeric data to assess climate conditions both locally and globally. French and Niculae [11] explored problems existing in current predictive models used in emergency management and suggest alternatives. Kararsova et al [19] also used association rule mining to determine the relationships spatially between fire and rescue incident locations. Papamichail and French [29] created a system that supports decision making in terms of emergencies in a nuclear power plant. Berndt et al. [4] demonstrated that data warehousing can be used to explore novel situations regarding bioterrorism as well as suggestions for investigations after an incident occurs. Berndt et al. [5] have also demonstrated that data warehousing can be used in supporting quality assurance related to medical care.

2.2 Natural Language Processing

The Internet era leaves us with a huge number of articles in text format. Most of these articles lack metadata, and therefore classification of text is an increasingly useful subdomain of general classification. Automatic text classification is useful in many fields, such as classifying product reviews [9], spam filtering [1], guiding financial investments [39], and, among others. In regards to the use of machine learning in text classification, support vector machines [17] as well as Naive Bayes [28] have long been used successfully to this end. In addition, the J48 decision tree algorithm has also proved to be useful in regards to text classification [45]. Jain et al. [16] demonstrated that text classification is even more successful when classifiers are combined using simple or weighted voting. Besides text classification, text and sentiment classification are another two important subfields of natural language processing. Entity extraction seeks to extract people, places, and other important entities from text in an unsupervised manner. Etzioni et al. [10] created the KnowItAll system which not only extracts entities but finds relations between the entities. Takeuchi et al [42] used support vector machines to extract medical terminology from medical journal abstracts. Miller et al. [25] also used entity extraction to find important people and places mentioned in broadcast news. In regards to sentiment classification, Pang et al. [28] demonstrated the successful classification of movie reviews into positive and negative through the use of support vector machines. Go, et al. [12] successfully classified tweets using tweets with emoticons as training data. Wan [44] used English corpora and machine translation to classify the sentiment of product reviews written in Chinese.

3 NLP-based Crisis Identification System

Our NLP (Natural Language Processing) based crisis identification system is trained using a collection of 80 articles from the New York Times. We will first give an overview of the system architecture, and then discuss the process of data collection and move on to the inner workings of the system itself.

3.1 System Architecture

The overall architecture of our news media article classification system is shown in Fig. 1. In the configuration file build.sbt, shown as STB in the figure, we specify all external dependencies by group ID, artifact ID, and version. When the application is executed, if a dependent library is not found locally (MySQL in the figure), the framework will search for and download the library remotely from the Maven Central Repository, which is a repository of build artifacts [2]. This allows the code to be more portable between machines by reducing the hassle in collecting external libraries.

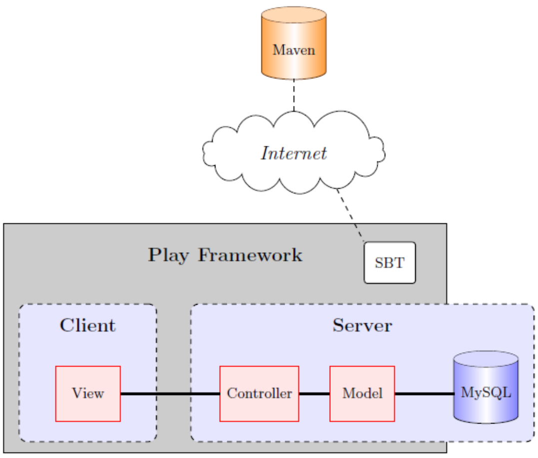

Fig. 1 System architecture model of the Play Framework integrating the various technical components

We use a simple Model-View-Controller design pattern for the separation of responsibilities using the integrating tool Play Framework [32]. The role of the Model is to translate the data in the database into Scala or Java data types. We define an Article class in the Model that stores all of the article meta data as String values. The View is strictly responsible for displaying and downloading results, such as word counts, classification results, and spreadsheets. The Controller is where the vast majority of the data mining process takes place.

Play Framework is a web application framework that is an integrated, easy-to-deploy web server, supports for Java's imperative and Scala's functional programming paradigms, supports native XML and JSON parsing, and provides a prototype-friendly templating library for creating HTML5 content quickly. These features are useful for the work presented in this paper.

3.2 Data Collection

In order to create a system which can identify new crises we must collect data for training. The New York Times Online Archive is utilized to facilitate the collection of crisis-related news media. The web interface for the archive allows users to search for articles by specifying a search term, date range, result type (article or blog), author, or section [43]. A total of 80 articles were collected about organizational crisis. The articles were downloaded as .html files and labeled manually as one of the three stages of the three-stage crisis model (see Table 1). Due to the nature of news media data, crises were only selected as training data if the organizations had news media exposure during all three stages in regard to a single crisis.

Table 1. Distribution of articles in training set

Class	Number of Articles
Pre-crisis	36
Crisis	18
Post-crisis	26
Total	80

The articles collected cover eight organizations that experienced a crisis between 2003 and 2013. An article is labeled with the pre-crisis class as long as the main topic of the article is not about a crisis that has occurred or is occurring. An article is labeled as crisis class if the news article is the first exposure of the crisis to the public through the news media. An article may also be in the crisis class if the organization has not yet contained the crisis or if the crisis is still expanding. An article is labeled as post-crisis if the impact of the crisis has been fully realized or if the organization is reacting to a crisis using a post-crisis communication strategy.

Per article, we collect the publication date of the article, the name of the organization involved, the text of the article, and its url. Listed below is a brief summary of each crisis:

(1) *Space Shuttle Columbia*: On February 2, 2003, the space shuttle Columbia disintegrated on re-entry into the earth's atmosphere, killing the seven astronauts on board [38].

(2) *BP Oil Spill:* On the evening of April 20, 2010, an explosion on the BP-owned oil rig Deepwater Horizon left 11 workers missing and later declared dead [36].

(3) *Apple-FoxCon:* Apple experienced negative publicity over its relationship with the Chinese electronic manufacturing company Foxconn when a factory explosion in May of 2011 killed two and injured over a dozen others [3].

(4) *News of the World:* was revealed in July of 2011 that employees of News of the World, a tabloid owned by Rupert Murdoch's News Corpora tion, had hacked the phones of British soldiers and of victims of violent crimes [3].

(5) *JP Morgan Chase Losses*:} In May of 2012, the multi-billion dollar financial institution JP Morgan Chase announced trading losses of over $2 billion USD [34]

(6) *Lance Armstrong Doping*: When the Unites States Anti-Doping Agency made public its evidence in a doping case against Lance Armstrong in the Fall of 2012, there were negative impacts for both his public image and for the charity that he co-founded, the Livestrong Foundation [24].

(7) *Exxon pipeline*: On March 29, 2013, Exxon Mobil's Pegasus pipeline ruptured near a small town in central Arkansas [40].

(8) *Target Data Breach*: From November 27 to December 15, 2013, the second largest breach of a retailer in history occurred, acquiring suc cessfully the personal information of customers of the American retail chain Target [31].

3.3 Data Preparation

Text Preparation

In early experiments with the crisis data, we found that the remove of named entities from the text greatly increased the accuracy of the crisis identification model. We found that the named entities led to overfitting for some articles at the expense of others. Therefore the first step in data preparation is the identification and removal of named entities. We used Stanford's Named Entity Recognizer (NER), for entity recognition. The parser is configured to identify seven classes of named entities: time, location, organization, person, money, percent, and date. All of the entities recognized by the parser are then removed. After all of the entities are removed the term frequency / inverse document frequency for all words are calculated using the Weka machine learning tool. This vector of tf-idf scores becomes the bag of words feature used in later classification. Bag of words refers to the method of treating each word as a feature without concern for each word's relation to other words in terms of ordering. The settings used to create the word vector in Weka are given below in Table [2].

Table 2 Parameters of the Weka StringToWordVector filter

Parameter	Value
IDFTransform	true
TFTransform	true
attributeIndices	(*the column of the article text*)
normalizeDocLength	normalize all data
stemmer	LovinsStemmer
useStoplist	true
wordsToKeep	1000000

Sentiment Classification

There are two sets of metrics that are being collected for sentiment scores. The first is a scalar metric of positive or negative sentiment that ranges from -5 to 5. The second set of metrics is a vector of six sentiment categories: positive, negative, litigious, uncertain, modal weak, and modal strong. Because all of the sentiment analysis is executed programmatically in our own code, we use the version of the articles that have been stemmed using the Porter stemmer Java library. We also apply the Porter stemmer to the dictionaries that contain the sentiment data for words. This allows us to look-up words from the articles in the dictionaries by their stems and to not have to worry about small variations in highly similar words. The first, scalar metric is calculated using Algorithm 1. We start by looking up every word of an article in the AFINN-111 dictionary map [26]. The word is the key in the map, and the value is the sentiment of the word, ranging from negative five to positive five. The sentiment of an article is calculated by summating the sentiments of the words. In order to normalize the sentiment of articles, we divide the total by the total number of words *N* in the article that were contained in the dictionary map. The result is the sentiment of the article from a scale of -5 to 5.

Algorithm 1: Calculate Scalar Sentiment

Input: A – an article
Input: D – AFINN dictionary map
Output: S – sentiment measure of A, a scale value from -5 to 5

begin
  $S \leftarrow 0$
  $N \leftarrow 0$
  foreach $word \in A$ **do**
    if $word \in D.keys$ **then**
      $S \leftarrow S + D[word]$
      $N \leftarrow N + 1$
  end
  $S \leftarrow S \div N$
  return S
end

The second, vector metric is calculated using Algorithm 2. We begin by looking up every word of an article in each of six word lists, one for each category of sentiment. The number of words in each category is totaled for an article. We start by looking up every word of an article in each of six word lists, one for each category of sentiment. The number of words in each category is totaled for an article. Then, the number of words in each category is divided by the number of words in any category. The result is the percentage of words that belong to each category of sentiment. Note that some words belong to multiple categories of sentiment, so the total of the percentages can exceed 100\%. For example, ``always" is both modal-strong and positive, and ``apparently" is both modal-weak and uncertain [22].

Algorithm 2: Calculate Categorical Sentiment as Vector

```
Input: A – an article
Input: D – list of dictionaries, one for each category
Output: V – vector of sentiment measures of A
begin
    V ← {0, 0, 0, 0, 0, 0}
    N ← 0
    foreach word ∈ A do
        for i = 0 to 5 do
            if word ∈ D[i] then
                V[i] ← V[i] + 1
                N ← N + 1
        end
    end
    for i = 0 to 5 do
        V[i] ← V[i] ÷ N
    end
    return V
end
```

3.4 Feature Selection

In this step we select the features which are most useful in classification as a form of feature selection. At this point we have 4,424 features, 4,417 of these features are tf-idf scores for individual stemmed words, and the remaining seven features consist of six scalar sentiment scores and one continuous sentiment score. We use the J48 decision tree implementation provided in the Weka [13] machine learning toolkit for feature selection. In a decision tree, the nodes with the lowest depth offer the most information gain. Since all of the features of our data are continuous variables, the nodes in the resulting decision tree contain a threshold for which a feature provides information. The final decision tree contains the features which provide the most useful information in the classification of articles, and features that provide redundant or no information are omitted from the decision tree. In the case of our data set, twelve word stems provided the most information in regards to classifying news articles: accis, advic, breach, commander, deb, den, emerg, encrypt, handl, hear, review, and widespread. In Fig. 2, we can see the values for which the TF-IDF scores were considered useful for these features, and in Table 3 we can see the words which each stem represents.

```
J48 pruned tree
------------------

advic <= 0
|   deb <= 0
|   |   encrypt <= 0
|   |   |   den <= 1.506716
|   |   |   |   commander <= 1.609757
|   |   |   |   |   emerg <= 1.295889
|   |   |   |   |   |   breach <= 0
|   |   |   |   |   |   |   accis <= 0
|   |   |   |   |   |   |   |   hear <= 1.115454
|   |   |   |   |   |   |   |   |   widespread <= 0.701946
|   |   |   |   |   |   |   |   |   |   handl <= 0
|   |   |   |   |   |   |   |   |   |   |   review <= 0:  pre crisis (36.0)
|   |   |   |   |   |   |   |   |   |   |   review > 0:  post crisis (2.0)
|   |   |   |   |   |   |   |   |   |   handl > 0:  post crisis (2.0)
|   |   |   |   |   |   |   |   |   widespread > 0.701946:  post crisis (4.0)
|   |   |   |   |   |   |   |   hear > 1.115454:  post crisis (5.0)
|   |   |   |   |   |   |   accis > 0:  post crisis (9.0)
|   |   |   |   |   |   breach > 0:  post crisis (3.0/1.0)
|   |   |   |   |   emerg > 1.295889:  crisis event (3.0/1.0)
|   |   |   |   commander > 1.609757:  crisis event (3.0/1.0)
|   |   |   den > 1.506716:  crisis event (3.0)
|   |   encrypt > 0:  crisis event (4.0)
|   deb > 0:  crisis event (3.0)
advic > 0:  crisis event (3.0)

Number of Leaves  :     13

Size of the tree :      25
```

Fig. 2 Results of J48 decision tree analysis

Table 3. Reverse look-up of word stems

Stem	Potential Matches
accis	accident, accidental, accidentally
advic	advice
breach	breach, breached, breaches, breaching
commander	commander, commanders
deb	debatable, debate, debated, debates, debating
den	den, denature, denial, denials, denied, denier, deniers, denies, dens, deny, denying
emerg	emerge, emerged, emergence, emergencies, emergency, emergent, emerges, emerging
encrypt	encrypt, encrypted, encryption, encrypts
handl	handle, handled, handles, handling
hear	hear, hearing, hearings, hears
review	review, reviewed, reviewing, reviews
widespread	widespread

3.5 Classification: Naive Bayes Model

In this section, we measure the performance of applying Weka's Naive Bayes classifier to the features extracted from the New York Times articles relating to crises. We compare the results of applying the Naive Bayes classifier on all 4,424 features to the results of applying the Naive Bayes classifier on the twelve text features selected from the J48 decision tree analysis, and the results of applying the Naive Bayes classifier to the twelve selected text features plus the sentiment features. The three models are evaluated by the true positive rate, false positive rate, precision, recall, and receiver operating characteristic (ROC Curve). The weighted averages are based on the distribution of the data set that belongs to each class. The target class is pre-crisis, crisis, or post-crisis and is judged on a per document basis rather than a topic basis.

The confusion matrix for all three Naive Bayes models, as well as the precision recall and F1 measures are given in Tables 4(a),4(b), and 4(c).

Table 4. Confusion matrices for the three Naive Bayes models

(a) All features

		Predicted		
		A	B	C
Actual	A	32	2	2
	B	5	8	5
	C	7	5	14

(b) Selected features with sentiment

		Predicted		
		A	B	C
Actual	A	35	1	0
	B	1	15	2
	C	0	2	24

(c) Selected features only

		Predicted		
		A	B	C
Actual	A	35	1	0
	B	1	16	1
	C	1	13	32

Table 5. Classification accuracy measures

(a) Pre-crisis article classification

Attribute	All Features	Selected Features
TP Rate	88.9%	97.2%
FP Rate	27.3%	2.3%
Precision	72.7%	97.2%
Recall	88.9%	97.2%
ROC Area	80.6%	97.8%

(b) Crisis article classification

Attribute	All Features	Selected Features
TP Rate	44.4%	83.3%
FP Rate	11.3%	4.8%
Precision	53.3%	83.3%
Recall	44.4%	83.3%
ROC Area	69.0%	90.3%

(c) Post-crisis article classification

Attribute	All Features	Selected Features
TP Rate	53.8%	92.3%
FP Rate	13.0%	3.7%
Precision	66.7%	92.3%
Recall	53.8%	92.3%
ROC Area	72.1%	92.4%

(d) Weighted averages

Attribute	All Features	Selected Features
TP Rate	67.5%	92.5%
FP Rate	19.0%	3.3%
Precision	66.4%	92.5%
Recall	67.5%	92.5%
ROC Area	75.2%	94.4%

When we take a closer look at the detailed accuracy of the classification models, we can see the metrics supporting the initial impression of the confusion matrices. The standard measures of the classification accuracy are given in Table [5]. The measures for pre-crisis, crisis, and post-crisis are given in Tables 5(a), 5(b), and 5(c), and the weight averages are given in Table 5(d). In the model with all of the features present, the weighted average percentage of correctly classified instances is 67.5\%, just over two thirds (see Table 5(d)). While this is about twice the accuracy of randomly guessing when given three classes, there is still plenty of room for improvement. The impression of the model with all 4,424 features worsens when we look at the true positive rate for the individual classes. Because there are more pre-crisis articles than any other class, the higher true positive rate for pre-crisis articles has the most influence on the weighted average. In the model with selected features, not only is the weighted average of true positive rate much higher at 92.5\% (Table 5(d)), but the true positive rate varies much less between the three classes (97.2\%, 83.3\%, and 92.5\%, respectively). This accuracy was calculated using 5-fold cross validation All other metrics in the model with the selected features improve as well: the false positive rates lower, precision and recall increase, and the ROC area increases for all three classes as shown in Fig. 3(a), 3(b) and 3(c).

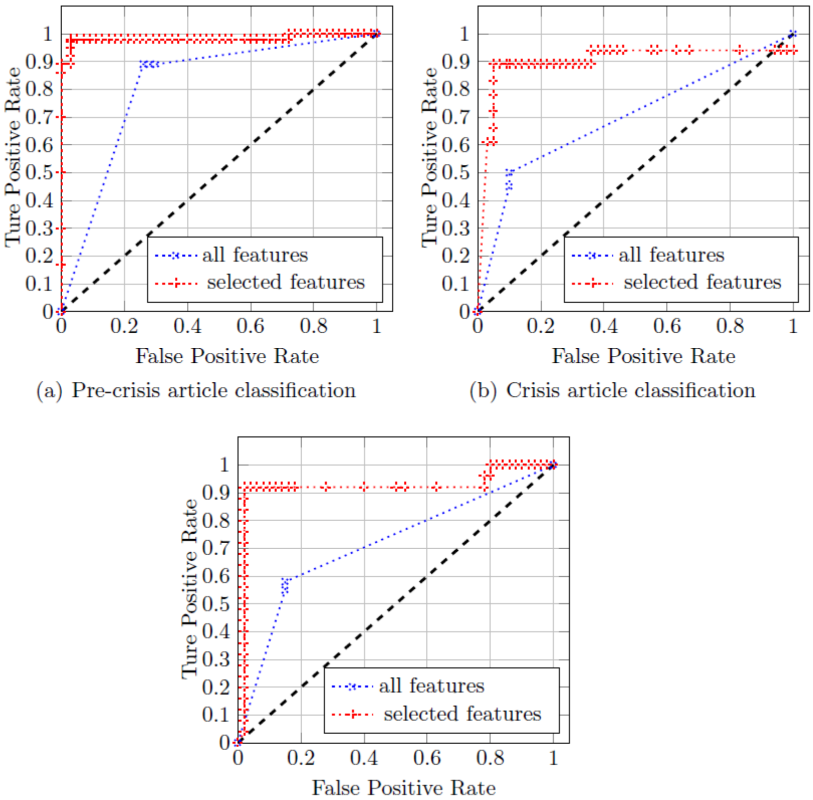

Fig. 3 ROC curves

4 Conclusion

In this paper, we set out to use data mining and machine learning techniques to create a reusable methodology that can be applied towards the empirical research of crisis communication. Because research in crisis communication has been dominated by case studies, our initial effort in this area has been to create a corpus of news media data that is labeled with characteristics of crisis communication theory. We aggregated and labeled news media data from the New York Times Article Archive, and we were successfully able to train a classification model to recognize the occurrence of specific terms in order to identify characteristics of organizational crisis. We also explored how our corpus and data mining process may be integrated into a multi-tier web application to be utilized by researchers on the World Wide Web. Though incident management systems do exist our incident management system is built on a major theory of crisis communication, the Coombs model. In addition, our system requires no data preparation, as it can classify the text of any article, whereas again other systems often require the use of numeric data which must be prepared before use. In regards to future work we would like to perform several activities in regards to data collection and machine learning. First, we plan to collect data other than news stories from New York Times. We worry that the wording used in New York Times might not map well to other news sources and might result in an overfit model. In addition, we would like to collect more data so it is possible to perform machine learning on a per topic basis in addition to a per document basis. However we demonstrate that our system is effective on several different crisis types and with the collection of more data across news sources could be especially powerful. In summary, our work focuses on a natural language based real time crisis identification system which requires no data preparation by the user and has already been evaluated on several existing crises.

References

1. Androutsopoulos, I., Koutsias, J., Chandrinos, K.V., Paliouras, G., Spyropoulos, C.D.: An evaluation of naive bayesian anti-spam filtering. arXiv preprint cs/0006013 (2000)
2. Apache: Apache Maven Project. http://maven.apache.org/index.html
3. Barboza, D.: Explosion at Apple Supplier Caused by Dust, China says. http: //www.nytimes.com/2011/05/25/technology/25foxconn.html (2011)
4. Berndt, D.J., Fisher, J.W., Craighead, J.G., Hevner, A.R., Luther, S., Studnicki, J.: The role of data warehousing in bioterrorism surveillance. Decision Support Systems 43(4), 1383{1403 (2007)
5. Berndt, D.J., Fisher, J.W., Hevner, A.R., Studnicki, J.: Healthcare data warehousing and quality assurance. Computer 34(12), 56{65 (2001)
6. Chen, H., Chung, W., Xu, J.J., Wang, G., Qin, Y., Chau, M.: Crime data mining: a general framework and some examples. Computer 37(4), 50{56 (2004)
7. Coombs, W.T.: Ongoing Crisis Communication: Planning, Managing, and Responding. Sage Publications, Thousand Oaks, CA, USA (1999)
8. Coombs, W.T.: Protecting Organization Reputations During a Crisis: The Development and Application of Situational Crisis Communication Theory. Corporate Reputation Review 10(3), 163{176 (2007)
9. Dave, K., Lawrence, S., Pennock, D.M.: Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In: Proceedings of the 12th international conference on World Wide Web, pp. 519{528. ACM (2003)
10. Etzioni, O., Cafarella, M., Downey, D., Popescu, A.M., Shaked, T., Soderland, S., Weld, D.S., Yates, A.: Unsupervised named-entity extraction from the web: An experimental study. Artificial Intelligence 165(1), 91{134 (2005)
11. French, S., Niculae, C.: Believe in the model: mishandle the emergency. Journal of Homeland Security and Emergency Management 2(1) (2005)
12. Go, A., Bhayani, R., Huang, L.: Twitter sentiment classification using distant supervision. CS224N Project Report, Stanford pp. 1{12 (2009)
13. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA data mining software: an update. SIGKDD Exploration Newsletter 11(1), 10{18 (2009)

14. Harms, S.K., Deogun, J., Saquer, J., Tadesse, T.: Discovering representative episodal association rules from event sequences using frequent closed episode sets and event constraints. In: Proceedings IEEE International Conference on Data Mining (ICDM 2001), pp. 603{606. IEEE (2001)
15. Jain, G., Ginwala, A., Aslandogan, Y.A.: An approach to text classification using dimensionality reduction and combination of classifiers. In: Information Reuse and Integration, 2004. IRI 2004. Proceedings of the 2004 IEEE International Conference on, pp. 564{569. IEEE (2004)
16. Joachims, T.: Text categorization with support vector machines: Learning with many relevant features. Springer (1998)
17. Jones, K.S.: A statistical interpretation of term specificity and its application in retrieval. Journal of documentation 28(1), 11{21 (1972)
18. Karasova, V., Krisp, J.M., Virrantaus, K.: Application of spatial association rules for improvement of a risk model for pre and rescue services. Proceedings of ScanGIS2005 (2005)
19. Lee, D.H., Jeng, S.T., Chandrasekar, P.: Applying data mining techniques for tracincident analysis. Journal of The Institution of Engineers, Singapore 44(2) (2004)
20. Liang, X.: Government Crisis Communication on the Microblog: A Theory Framework and the Case of Shanghai Metro Rear-end Collision. In: Proceedings of the 6th International Conference on Theory and Practice of Electronic Governance, ICEGOV '12, pp. 248{257. ACM (2012)
21. Loughran, T., McDonald, B.: When is a liability not a liability? Textual analysis, dictionaries, and 10-Ks. The Journal of Finance 66(1), 35{65 (2011)
22. Lyall, S.: British Tabloid Hacked Missing Girl's Voice Mail, Lawyer Says. http: //www.nytimes.com/2011/07/05/world/europe/05britain.html (2011)
23. MacLaggan, C.: Exclusive: Livestrong cancer charity drops Lance Armstrong name from title. http://tinyurl.com/zrx9slu (2012)
24. Miller, D., Schwartz, R., Weischedel, R., Stone, R.: Named entity extraction from broadcast news. In: Proceedings of DARPA Broadcast News Workshop, pp. 37{40 (1999)
25. Nielsen, F.A.: AFINN. http://www2.imm.dtu.dk/pubdb/p.php?6010 (2011) Nielsen, F.A.: A new ANEW: Evaluation of a word list for sentiment analysis in microblogs. arXiv preprint arXiv:1103.2903 (2011)
26. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up?: sentiment classification using machine learning techniques. In: Proceedings of the ACL-02 conference on empirical methods in natural language processing-Volume 10, pp. 79{86. Association for Computational Linguistics (2002)
27. Papamichail, K.N., French, S.: Design and evaluation of an intelligent decision support system for nuclear emergencies. Decision Support Systems 41(1), 84{111 (2005)
28. Peng, Y., Zhang, Y., Tang, Y., Li, S.: An incident information management framework based on data integration, data mining, and multi-criteria decision making. Decision Support Systems 51(2), 316{327
29. Perlroth, N.: Target's Nightmare Goes On: Encrypted PIN Data Stolen. http://bits.blogs.nytimes.com/2013/12/27/ targets-nightmare-goes-on-encrypted-pin-data-stolen (2013)
30. Play Framework: Play 2.1 documentation. http://www.playframework.com
31. Porter, M.F.: An algorithm for sux stripping. Program: electronic library and information systems 14(3), 130{137 (1980)
32. Reuters: Banks Suer After-Hours On Word of Chase Losses. http://www. nytimes.com/2012/05/11/business/daily-stock-market-activity.html (2012)
33. Rish, I.: An empirical study of the naive Bayes classifier. In: IJCAI workshop on empirical methods in artificial intelligence, vol. 3(22), pp. 41{46 (2001)

34. Robertson, C.: Search Continues After Oil Rig Blast. http://www.nytimes.com/2010/04/22/us/22rig.html (2010)
35. Safavian, S.R., Landgrebe, D.: A survey of decision tree classifier methodology. IEEE transactions on systems, man, and cybernetics 21(3), 660{674 (1991)
36. Sanger, D.E.: Loss of the Shuttle: the Overview; Shuttle Breaks Up, 7 Dead. http://www.nytimes.com/2003/02/02/us/loss-of-the-shuttle-the-overview-shuttle-breaks-up-7-dead.html (2003)
37. Schumaker, R.P., Chen, H.: Textual analysis of stock market prediction using breaking financial news: The azn text system. ACM Transactions on Information Systems (TOIS) 27(2), 12:1{12:19 (2009)
38. Schwirtz, M.: Oil Pipeline Ruptures in Arkansas. http://www.nytimes.com/2013/03/31/us/oil-pipeline-ruptures-in-arkansas.html (2013)
39. Seeger, M.W., Sellnow, T.L., Ulmer, R.R.: Communication and Organizational Crisis. Praeger Publishers, Westport, CT, USA (2003)
40. Takeuchi, K., Collier, N.: Bio-medical entity extraction using support vector machines. Artificial Intelligence in Medicine 33(2), 125{137 (2005)
41. The New York Times: New York Times Article Archive. http://www.nytimes. com/ref/membercenter/nytarchive.html (2014)
42. Wan, X.: Using bilingual knowledge and ensemble techniques for unsupervised chinese sentiment analysis. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 553{561. Association for Computational Linguistics (2008)
43. Youn, S., McLeod, D.: A comparative study for email classification. In: Advances and Innovations in Systems, Computing Sciences and Software Engineering, pp. 387{391. Springer (2007)