A Tweets Mining Approach to Detection of Critical Events Characteristics using Random Forest

CARLOS ENRIQUE GUTIERREZ and MOHAMMAD REZA ALSHARIF
Department of Information Engineering, University of the Ryukyus, Okinawa, Japan
KATSUMI YAMASHITA
Graduate School of Engineering, Osaka Prefecture University, Osaka, Japan
and
MAHDI KHOSRAVY
University for Information, Science and Technology, “Saint Paul the Apostle”, Ohrid, Macedonia

During a natural disaster, while most people are overwhelmed, governmental agencies are in charge of public safety, and they must timely provide true directions, by quick and efficient analysis of massive amount of information. To guaranty decision making process, it is needed to extract and organize most important information as soon as possible to avoid adding more confusion. In this paper we show an automatic analysis of complex data such as text generated during an emergency event. An unsupervised and recent popular machine learning model called random forest is trained to uncover and organize predominant features from a large set of tweets; providing a hierarchy of main variables which might indicate rules and an approximation of how information flows during an unusual event. In our work we provide, firstly, a conversion process from text to numerical vectors for the training data; secondly, we introduce briefly random forest model; next, we expose an algorithm to adapt random forest to our problem; finally we show results for different configuration of the model and conclusions.

Keywords: Random forest, main features detection

1. INTRODUCTION

During and after a natural disaster on-line newspaper, social networks and blogs become very active describing many situations interrelated. On internet, people start talking about concerns, worries, and several other aspects that deserve to be analyzed. The need to improve our chances of surviving forces us to study and learn more about the main characteristics of an emergency situation. In this paper we focus on the adoption of random forest model to detect and extract from a large set of tweets main features and related rules of a sudden critical event. After a number of several applications, mainly for computer vision [Nowozin et al. 2011] [Bosch et al. 2007], its performance has been proved to achieve excellent results. Random forest's generalization accuracy on unseen data is verified at [Ho 1995]; and its full randomization is demonstrated to be computationally efficient at [Geurts et al. 2006]. A unified random decision forest model has been presented at [Criminisi et al. 2012] for classification, regression, density estimation, manifold learning, semi-supervised learning, and active learning. Our application uses a density estimation model with unsupervised learning to detect main features and split the data automatically. A set of predominant features is an approach to and might represent a behavioral pattern and concerns of the affected users community, town, city, country. The problem of main features extraction from a large set of text sources is a large data mining problem, where a vast amount of text require to be processed immediately. Random forest has the advantage to take a huge amount of data, split it in small sub-sets and assign them to individual decision trees. Trees process the information in parallel, and after a relatively short time of learning, outputs are combined to

Author's address: C.E Gutierrez, Graduate School of Engineering and Science, University of the Ryukyus, 1 Senbaru, Nishihara, Okinawa, Japan.

provide a final and unified result.

2. DATA SET AND EMERGENCY EVENT

The real world critical events are reflected on Twitter by an exponential growth of posts or comments by the users with a direct relation to that particular event. We recorded tweets during a minor earthquake in California on 8th May 2014; people felt the earthquake and immediately reported brief text updates on Twitter. This situation is well recognized by analyzing figure 1; it shows how at 06:41 UTC time the number of tweets per minute increases rapidly, returning after 25 minutes approximately to an average number. Our paper focuses on that 25 minutes time interval to extract information, where a huge amount of tweets were created as an instantaneous reaction to a potential emergency.

Tweets are free text micro blogging posts of no more than 140 characters, used by millions of people around the world; with one important characteristic, its real-time nature. Although their length per post is limited, the variety of words that can be used is high. If we take in account that each single word represents a different variable, a tweet is considered a high dimensional data. Fortunately, groups of variables often move together; our previous work at [Gutierrez et al.], [Gutierrez et al. 2012] and [Gutierrez et al. 2013] exposed that, on text, more than one variable is measuring the same driving principle governing the systems behavior.

2.1 Numerical Representation of Tweets

Firstly, a dictionary of words was created from: the IPCC special report on *Managing the Risks of Extreme Events and Disasters to Advance Climate Change Adaptation* [Field 2012] and the *Terminology on Disaster Risk Reduction* of the United Nations [ISDR 2009]. The dictionary is a compound of the most influential terms required to evaluate an emergency situation sorted by the frequency found at the mentioned documents, which gives us a relative weight for each term. In addition, we created a second dictionary composed by most frequent words found on tweets that contain terms such as *tsunami, earthquake, quake, flood, cyclone, avalanche, blizzard, landslide, typhoon, etc.*

Having a dictionary, each tweet is processed on the fly as follows:

(1) Special characters, numbers, symbols, and meaningless words such as conjunctions, prepositions and adverbs were removed.

(2) Porter stemming process [Porter 1980] is applied; stemming is the process for reducing inflected or sometimes derived words to their stem, base or root form. The general idea underlying stemming is to identify words that are the same in meaning but different in form by removing suffixes and endings; for instance, words such as "caused, "causing" are reduced to the root word cause. Both dictionaries are composed by roots terms.

(3) Random forest selects from the dictionaries words at random and transforms a tweet in a reduced dimension vector where each element is a number equal to the frequency found for each term. We chose to have 2 representations of each tweet to compare results using different dictionaries.

The result is an unlabeled collection of numerical vectors used to train individual decision trees independently and in parallel; next section provides an introduction to random forest model.

3. DECISION TREES AND RANDOM FOREST

Decision trees have been studied for long time, many algorithms have been created to get a trained optimal decision tree, mainly for regression and classification [Breiman et al. 1984]. A decision tree is a classifier expressed as a recursive partition of the input space. The decision tree consists of a node called "root", "internal" or "test" nodes, and "leaves" or "terminal" nodes. Each test node split the input space into two or more sub-spaces according to a test function of the input attributes values.

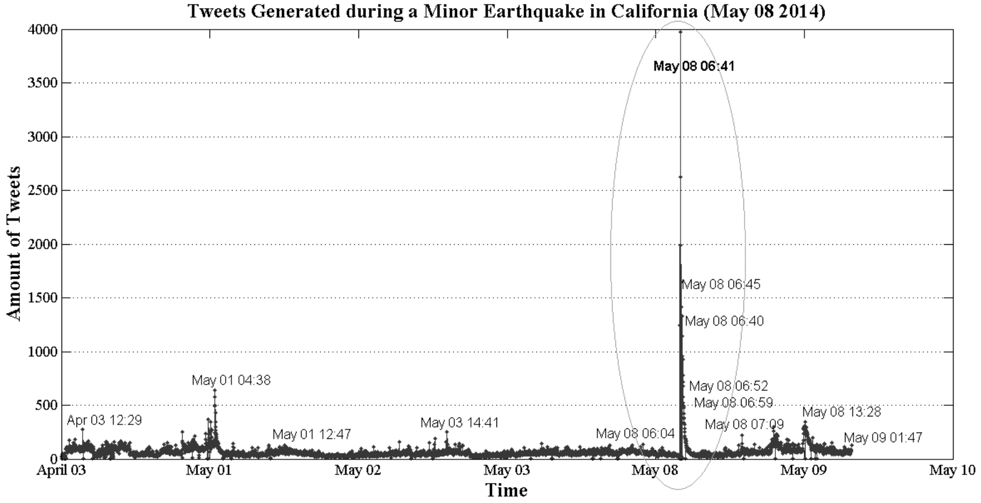


Figure 1: Users reaction on Twitter for a M3.3 Earthquake at Greater Los Angeles Area, California, May 8th 2014

On the contrary, random forests are an active research field in the present years, used for classification, regression, pattern recognition and density estimation. One of its famous successful implementations is the Microsoft kinect for XBox 360. A random decision forest is an ensemble of randomly trained decision trees, where all tree outputs are combined by averaging their class posteriors. This combination is the argumentation of random forests popularity; on classification tasks they produce much higher accuracy on previously unobserved data, improving substantially its generalization. Lets denote a generic data point by a vector $v = (x_1, x_2, x_3, ..., x_d) \in R^d$, where each x_i is a measurement of the i^{th} feature and d is the dimensionality. In most of the problems, the dimensionality of input vectors is high; however, in random forest it is not necessary to compute all d dimensions of v, they are randomly sampled from the set of all possible features by a function $\phi(v) : R^d \rightarrow R^{d'}$ with $d' << d$.

Random forest training consists of optimizing the parameters of the test functions for each internal node, in order to maximize an acquired *energy function.* Test functions split the data in different sets where each set has and associated *Shanon* entropy defined by:

$$H(S) = -\sum_{c \in C} p(c) \log(p(c)) \tag{1}$$

Where S is the training data set, and c is a category from the set of all defined categories C in case of labeled data. Having the entropy, the information gain is solved mathematically as:

$$I = H(S) - \sum_{i \in \{Left, Right\}} \frac{|S^i|}{|S|} H(S^i) \tag{2}$$

S is better class separated when the information gain is higher; the information gain is the *energy function* used to optimize the test function that produces the highest confidence. Each internal node j is associated with a binary test function denoted as:

$$h(v, \theta_j) \in \{0, 1\} \tag{3}$$

Where 0 and 1 indicate respectively “false” and “true”, input data is evaluated and sent to left or right according to the output of function above. In a general model $\theta_j = (\phi, \beta, \tau)$, where ϕ is a function that selects some features out of the entire set of features, β defines a geometric primitive used to separate the data, and τ are the thresholds used for the inequalities in test functions. For most of the cases, equation 3 is reduced to pick the best variable or split point among the d' features, which maximizes the information gain. Training a tree is achieved by optimizing each internal node test function parameters by:

$$\theta_j^* = \arg\max I_j \tag{4}$$

Randomness is essential in this method, individual component trees provides a de-correlated prediction; this characteristic helps to improve robustness with respect to noisy. Randomness is included during training by:

(1) bagging: random selection of the training data.

(2) randomized node optimization: from the entire set of all possible features we chose randomly a subset of θ.

During training, information that is useful for prediction is learned for all nodes. In classification, each leaf stores the empirical distribution over the classes associated to the subset of training data that has reached that leaf. In a forest with T trees where $t \in \{1, 2, ..., T\}$, trees combine their outputs into a single forest prediction by an averaging operation as follows:

$$p(c|v) = \frac{1}{T}\sum_{t=1}^{T} p_t(c|v) \tag{5}$$

After the above mentioned brief introduction to random forest for its most common application "classification", next section explains our implementation using unlabeled data with unsupervised learning, also known as clustering forest or density forest.

4. TWEETS MINING

We trained a random forest with $T = 100$ trees and deep $D = 4$ by using algorithm below:

```
(1) Repeat until complete $T$ trees:
    (a) Query on Twitter using keywords *natural disaster, tsunami, earthquake, quake, flood, etc.*
    (b) Assign query's results to set $S$ with size $N$ and create $i : 1$ to 20 trees by the following steps:
        i. Perform *bagging*, by drawing uniformly at random a sample $S_i$ of size $M < N$ from $S$.
        ii. Grow a decision tree $t_i$ by recursively repeating steps below for each internal node:
            A. Perform randomized node optimization by selecting $d'$ variables at random from $d$.
            B. Transform sampled tweets into numerical $d'$-dimensional vectors.
            C. Pick the best variable among the $d'$ variables. The best split feature maximizes the information gain for the node under training.
            D. Split the node into two child nodes.
            E. If the maximum deep is reached (parameter $D$), close tree $t_i$ and create next tree.
(2) Group split variables learned at each node for each tree and build a histogram of main variables.
```

Although the proposed algorithm is shown as "sequential", it is a completely distributed algorithm. It is possible to assign a query's result to some trees in a machine, another subset of data to trees in another machine, an so on. In today's world, when data sets are too large, it is unwise to load all data into memory. Even more, any emergency situation produces a flood of data to be processed. Our proposed algorithm takes several queries and split their results in small subsets. This model can be applied to process streams in real-time; an online random forest for classification has been proposed at [Saffari et al. 2009] that takes ideas from [Geurts et al. 2006] and implements an on-line decision tree growing procedure. Our work doesn't implement an on-line growing strategy, it manages the streaming data by querying Twitter at different intervals of time.

We apply Breiman's approach [Breiman 2001] that introduces a way of injecting randomness in the forest by randomly sampling the training data. As well, the technique is known as *bagging*. Trees takes at random a subset containing approximately a 10% of query's result. Nodes were trained by using a subset of features of interests randomly selected based on their weights at the dictionary.

We trained each tree searching intensely the best split variable for each generated subset, resulting in maximizing the information gain. Unlike classification, where it is possible to use equation 1 to obtain the entropy, we employed an unsupervised form of entropy. If we consider that each subset that reaches a node is explained by a multivariate *Gaussian* distribution, then the differential entropy of a d-variate *Gaussian* is defined as:

$$H(S) = \frac{1}{2}\log\left((2\pi e)^d|\Lambda(S)|\right) \tag{6}$$

Where $\Lambda(S)$ is a d x d covariance matrix and $|\Lambda(S)|$ is its determinant. Being d high; covariance matrix calculation returns values close to zero or zero, causing error and small negative values for the determinant. This problem is solved in our implementation by the application of two strategies:

(1) Reducing d by randomized node optimization.

(2) Adding a small bias λ to the diagonal of the covariance matrix before computing its determinant. We perform $\Lambda(S) + \lambda I$, where I is the identity matrix.

The information gain based on the unsupervised entropy is expressed as:

$$I_j = \log\left(|\Lambda(S_j)|\right) - \sum_{i\in\{Left,Right\}} \frac{|S_j^i|}{|S_j|}\log\left(|\Lambda(S_j^i)|\right) \tag{7}$$

Where $|.|$ indicates the determinant or the cardinality of the subsets. By optimizing previous equation, the process results in a tree that splits a subset into several clusters which are assumed to have a *Gaussian* distributions.

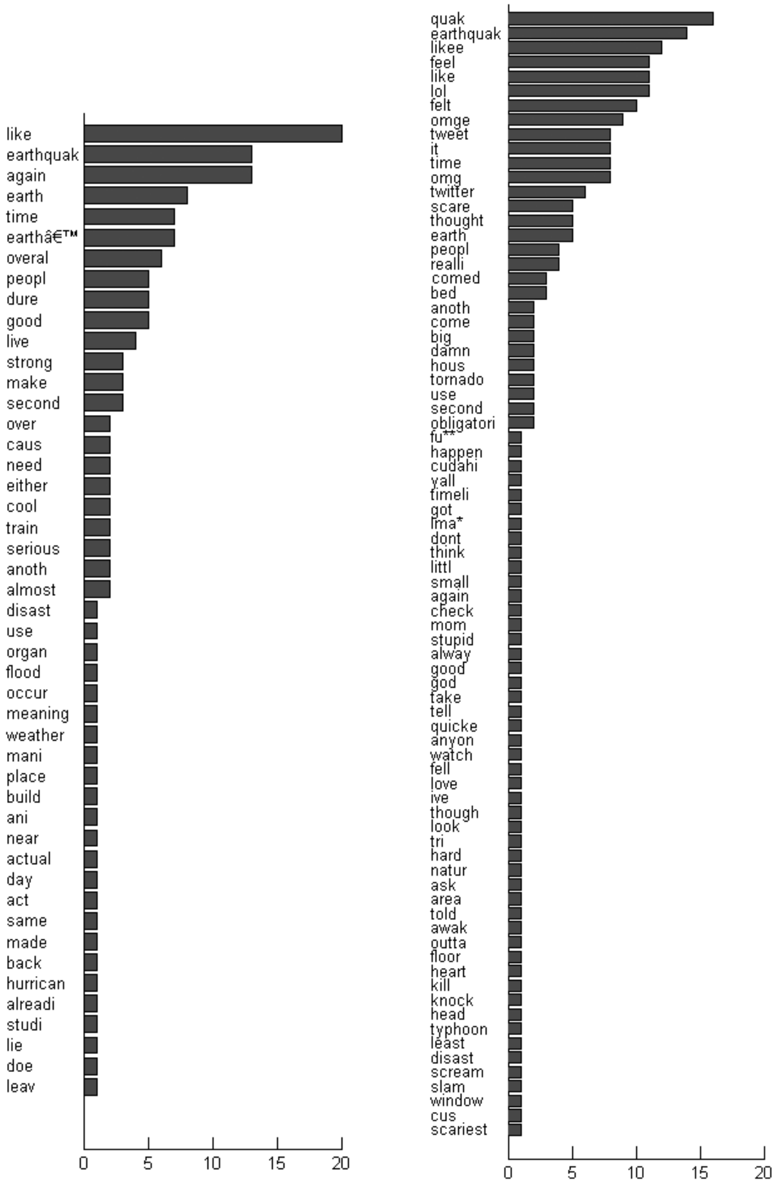
Our applications goal is not to analyze the generated clusters, neither to study any way to perform prediction. Instead, it searches to identify what is the information most useful to characterize the entire emergency data set. Split features play an important role; they contain the information needed to split training data set into a number of compact clusters. Therefore, for our application, the set of split features constitutes the set of main characteristics detected.

5. RESULTS

We trained two random forests with $T = 50$ trees and deep $D = 4$, with different dictionaries as mentioned previously. The selection of D is tied to amount of main variables we wish to obtain, higher values of D implies higher processing time. In each case, a bunch of features is presented and the model selects some of them as nodes in the trees. Selected features essentially "decide" on the data set; they are the best choices to provide information about an unsupervised clustering that minimizes the entropy about the emergency event. Main features help to assess the emergency by answering, for example, what is the best to ask? what is the best to evaluate?. After training, each tree generate a matrix F_{t_i} of d x $(2^{D+1}-1)$ (features x nodes), where element $F_{t_i}(k,j)$ is equal to 1 if feature k is the split variable of node j, otherwise 0. Main variables are calculated as follows:

$$Histogram(k) = \sum_{j=1}^{(2^{D+1}-1)} \sum_{i=1}^{T} F_{t_i}(k,j) \tag{8}$$

Equation 8 basically tells us the importance of each word when decisions are made on an

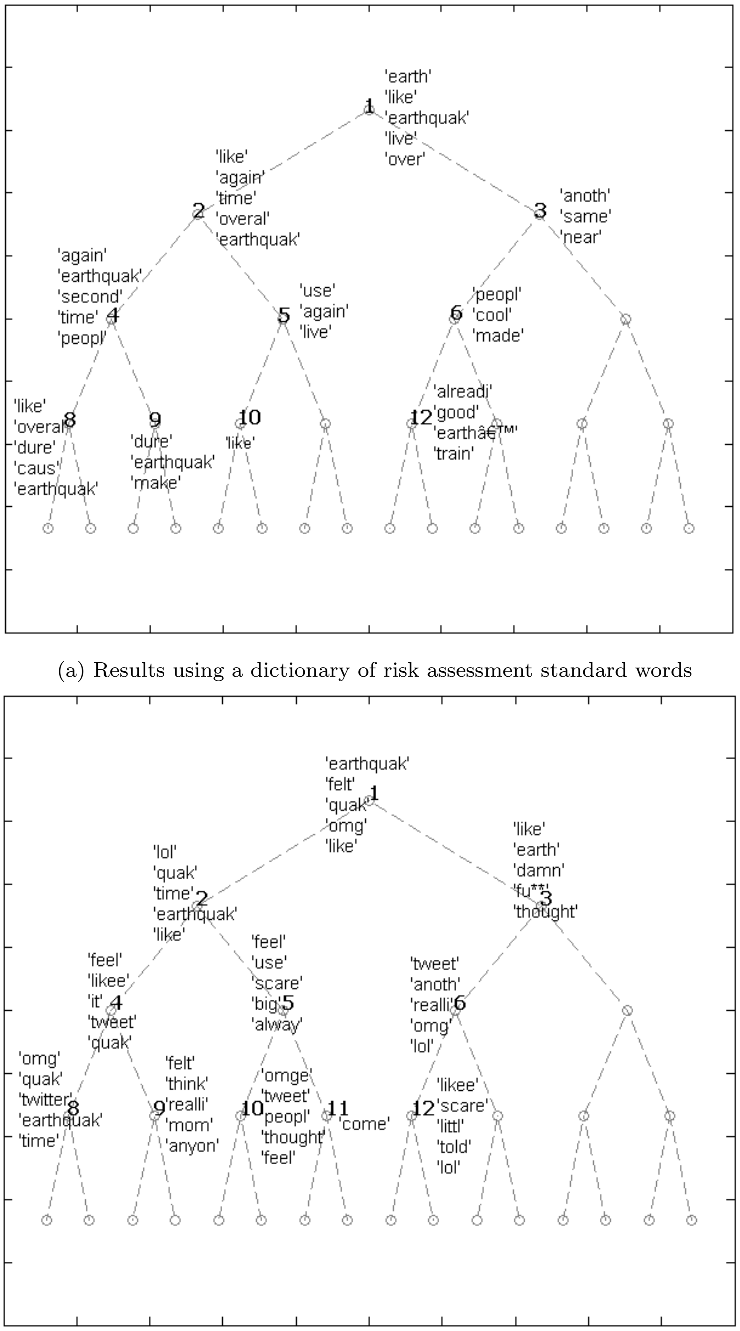


(a) Results using a dictionary of risk assessment standard words

(b) Results using a dictionary of common terms used in Twitter

Figure 2: Random Trees main features detection

emergency event; originally we have many features at each dictionary, now we have discovered naturally the most important characteristics in a reasonable amount of time. Figure 2 shows different features discovered, 1^{st} case gives a formal evaluation on standards {*like, earthquake, again, earth, time, people, strong, serious, ...*}, while the 2^{nd} set of words shows people's reactions by common terms {*earthquake, feel, lol, omg, scare, damn, ...*}. This comparison gives us an idea of how important is the data dictionary. For similar situations in the future, it is essential to have as many resources as possible available. Any assessment system should include for critical regions, for example countries located on edges of tectonic plates, dictionary of locations, list of first and last names, stations names, landmarks names, list of rivers, schools, hospitals, public



(a) Results using a dictionary of risk assessment standard words

(b) Results using a dictionary of common terms used in Twitter

Figure 3: Hierarchy of main features for rules detection

offices, geopolitical entities, etc. Our system was unable to identify locations from tweets due the lack of features at used dictionaries. Complete dictionaries are essential for a larger scale event in future situations. However, in a more abstract level, we are showing that random forest model has the potential to make an important contribution in a disaster response situation.

At a real emergency, after processing and averaging some few trees, we might have an approximation to assess the situation. Over the time, new trained trees are incorporated to improve the results.

Learned main features might be organized as rules; to detect rules we average the component trees of a random forest with $T = 100$ and deep $D = 4$, into one single tree that shows relations among variables. Figure 3 displays for each node the most predominant five split features derived from:

$$Node(j) = \underset{1<k<d}{\arg\max} \left(\sum_{i=1}^{T} F_{t_i}(k,j) \right) \tag{9}$$

Branches to the right are positive evaluation results of split features, while branches to the left are negative results. For example rules discovered for 1^{st} case using a dictionary of terms from risk assessment standards can be the sequence of nodes 1, 3, 6: *YES(earth, like, earthquake, live, over)* → *NO(another, same, near)* → *(people, cool, made)*, and the sequence of nodes 4, 9: *YES(again, earthquake, second, time, people)* → *(dure, earthquake, make)*.

While for the 2^{nd} case, using a dictionary of terms commonly used in Twitter, uncovered rules such as sequence 2, 5, 11: *YES(lol, quake, time, earthquake, like)* → *YES(feel, use, scare, big, always)* → *(come)* and sequence 4, 9: *YES(feel, like, it, tweet, quake)* → *(felt, think, really, mom, anyone)* provide information oriented to people's reactions.

Due that our data set corresponds to a small earthquake with no damage reported, the trees tend to organize the data to their left "negative" side. We believe that for real emergencies information will be spread mainly to the right side of the trees.

6. CONCLUSION

We have studied random forest and applied it to uncover main features during an anomalous event. Random forest automatically and with no previous information organized predominant terms in a ranking list that indicates their importance within the discovered set. Variables are the result of maximizing systematically the information gain, and their frequency shows and approximation of how information propagated during the emergency event. A averaged tree has been proposed showing the main test features for each node, from where rules can be detected.

We have compared results and the effect of using different dictionaries; the analysis of the words have shown how the emergency event can be described from different perspectives, being necessary the usage of multiple dictionaries of locations, names, an other entities for real situations. We believe that the empirical findings reported in this paper present a relevant model that can be used to provide information to decision takers during difficult times, and will play an important role as large-scale data processing model for discovering patterns.

REFERENCES

Bosch, A., Zisserman, A., and Munoz, X. 2007. Image classification using random forests and ferns.

Breiman, L. 2001. Random forests. *Machine learning 45,* 1, 5–32.

Breiman, L., Friedman, J., Stone, C. J., and Olshen, R. A. 1984. *Classification and regression trees.* CRC press.

Criminisi, A., Shotton, J., and Konukoglu, E. 2012. Decision forests: A unified framework for classification, regression, density estimation, manifold learning and semi-supervised learning. *Foundations and Trends® in Computer Graphics and Vision 7,* 2–3, 81–227.

Field, C. B. 2012. *Managing the risks of extreme events and disasters to advance climate change adaptation: Special report of the intergovernmental panel on climate change.* Cambridge University Press.

Geurts, P., Ernst, D., and Wehenkel, L. 2006. Extremely randomized trees. *Machine learning 63,* 1, 3–42.

Gutierrez, C., Alsharif, M., Cuiwei, H., Villa, R., Yamashita, K., Miyagi, H., and Kurata, K. 2012. Natural disaster online news clustering by self-organizing maps. ishigaki, japan. In *27th SIP symposium.*

Gutierrez, C., Alsharif, M., Villa, R., Yamashita, K., and Miyagi, H. Data pattern discovery on natural disaster news. sapporo, japan. *ITC-CSCC, ISBN*, 978–4.

Gutierrez, C. E., Alsharif, M. R., Cuiwei, H., Khosravy, M., Villa, R., Yamashita, K., and Miyagi, H. 2013. Uncover news dynamic by principal component analysis. *Shanghai, China, ICIC Express Letters 7,* 4, 1245–1250.

Ho, T. K. 1995. Random decision forests. In *Document Analysis and Recognition, 1995., Proceedings of the Third International Conference on.* Vol. 1. IEEE, 278–282.

ISDR, U. 2009. Unisdr terminology on disaster risk reduction. *Geneva, Switzerland, May*.

Nowozin, S., Rother, C., Bagon, S., Sharp, T., Yao, B., and Kohli, P. 2011. Decision tree fields. In *Computer Vision (ICCV), 2011 IEEE International Conference on.* IEEE, 1668–1675.

Porter, M. F. 1980. An algorithm for suffix stripping. *Program: electronic library and information systems 14,* 3, 130–137.

Saffari, A., Leistner, C., Santner, J., Godec, M., and Bischof, H. 2009. On-line random forests. In *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on.* IEEE, 1393–1400.

Carlos Enrique Gutierrez was born in Argentina. He received the B.Sc. from National University of Jujuy, Argentina in 2002, and M.Sc. degree from University of the Ryukyus, Japan in 2009. He is PhD candidate of Interdisciplinary Intelligent Systems at Department of Information Engineering, University of the Ryukyus, Okinawa, Japan. His research topics of interest are data mining, machine learning and big data.

Mohammad Reza Alsharif received the B.Sc. and M.Sc. degree in electrical engineering from the University of Tehran, in 1973 and 1974, respectively, and the Ph.D. degree in electrical engineering from the University of Tokyo in 1981. He was Head of Technical Department of IIRB College, Iran from 1981 to 1985. Then, he was a senior researcher at Fujitsu Labs. Co. Kawasaki, Japan from 1985 to 1992. From 1992 to 1997, he was an assistant professor in the school of electrical and computer engineering, University of Tehran, Tehran, Iran. From1997, Dr. Alsharif is a professor at the Department of Information Engineering, University of the Ryukyus, Okinawa, Japan. He has developed an algorithm and implemented its hardware for real time T.V. Ghost canceling. He introduced a new algorithm for Acoustic Echo Canceller and he released it on VSP chips. His research topics of interest are in the field of Blind Source Separation, MIMO Speech and Image Processing, MIMO Communication systems, Echo Canceling, Active Noise Control and Adaptive Digital Filtering. He is a senior member of IEEE, and a member of IEICE.

Katsumi Yamashita received the B.E. degree from Kansai University, the M.E. degree from Osaka Prefecture University and the Dr. Eng. degree from Osaka University in 1974, 1976 and 1985, respectively, all in electrical engineering. In 1982, he became an assistant professor in University of the Ryukyus, where he became a professor in 1991. Now he is a professor in Osaka Prefecture University. His current interests are in digital communication and digital signal processing. Dr. Yamashita is a member of the IEEE, IEICE, and IEEJ.

Mahdi Khosravy received the B.Sc. degree in electrical engineering from the Sahand University of Technology, Tabriz, Iran in 2002; M.Sc degree in biomedical engineering from Beheshti University of Medical Sciences, Tehran, Iran in 2004; and PhD degree of Interdisciplinary Intelligent Systems from University of the Ryukyus, Okinawa, Japan. Now he is assistant professor at University for Information Science and Technology, Ohrid, Macedonia. His research interest lies in the areas of Blind Source Separation, MIMO Speech and Image Processing, MIMO Communication systems, Linear and nonlinear Digital filters, Medical Signal and Image processing, ECG Preprocessing and ECG arrhythmia detection.