A Rule Based Feature Selection Approach for Target Classification in Wireless Sensor Networks with Sensitive Data Applications

Zhiyong Hao and Bin Liu

Science and Technology on Information Systems Engineering Laboratory, National University of Defense Technology, Changsha 410073, China

Correspondence should be addressed to Zhiyong Hao; haozhiyongphd@gmail.com

Received 18 October 2013; Revised 17 February 2014; Accepted 7 March 2014; Published 7 April 2014

Copyright © 2014 Z. Hao and B. Liu. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

One of the important issues faced in the domain of target classification in wireless sensor networks is the restricted lifetime of individual sensors, caused by limited battery capacity. Although the base station usually has sufficient energy supply and computational power, it is often deemed to be the object of enemy invading hostile terrain. Hence, minimizing energy consumption of sensors while maintaining a given classification accuracy is a key problem in this research area, especially for sensitive data applications. This paper proposes a rule based feature selection approach rather than all-features approach that aims at increasing the energy efficiency of the system without losing much classification accuracy. In experiments, the feasibility and effectiveness of our approach are demonstrated empirically.

1. Introduction

The field of collaborative target classification in wireless sensor networks (WSNs) community has become a focus of intensive research in recent years [1]. One of the important applications in this research area is target classification in hostile terrain, for example, identifying types of vehicles in battlefield. Generally, a WSN consists of a collection of nodes that can sense the environment using their local sensors and communicate with other nodes or a base station wirelessly. Due to restricted battery capacity, it is a main concern to minimize energy consumption in WSNs. Although the base station usually has sufficient energy supply and computational power, it is often deemed to be the object of enemy invading. In this case, the original historical data, which is known as the sensitive data, is usually not permitted to be stored on the base station. Hence, how to design an efficient target classification approach to minimize the energy consumption of sensors while maintaining a given classification accuracy is a key issue in this kind of application.

There are two main ways to minimize the energy consumption of a wireless sensor network based on different assumptions about the sensing range of sensors. One has been firstly investigated by Sun and Qi [2], to deal with the problem of target classification in energy-constrained wireless sensor networks. Their basic assumption is that each sensor is powerful enough to sense all the features of the target and can communicate with other sensor nodes arbitrarily. On the other hand, Csirik et al. [3] propose a sequential classifier combination approach for target classification based on the assumption that each sensor corresponds to exactly one feature of the target to be classified. Their main idea is to sequentially read features from sensors one by one for minimizing the energy consumption of sensors while achieving classification accuracy as high as possible. Our approach is closer to the latter but uses rule based classifier instead of instance based classifier like k-NN for sensitive data applications.

In this paper, we propose an alternative approach to all-features approach (that is measuring the target with all the sensors and transmitting every single acquired sensor reading to the base station) for target classification in WSNs with sensitive data application. It is a new feature selection approach. We address the feature selection problem
from a particular perspective, namely, in context of energy-constrained wireless sensor networks rather than the feature subset selection in most of the previous papers [4, 5], in which the objective was to find a subset of features that lead to a high classification accuracy. It is assumed in the current paper that there is only one base station on which the rule based classifier runs, rather than an instance based classifier, for example, k-NN classifier. Thus, there is no original historical data that need to be stored on the base station. In order to minimize the energy consumption of a WSN, we assume that sensors are only activated upon request from the base station according to the rule based classifier and each sensor is only corresponding to exactly one feature to be used for target classification. Thus, minimizing the energy consumption becomes equivalent to minimizing the number of features to be used, which is the point how our energy-efficient system works.

The rest of this paper is organized as follows. Section 2 gives background information on the rule based classifier and information theoretic principles for Prism family of algorithms used as the base classifier in our system. Section 3 presents FSMRC, a rule based feature selection approach to minimize the energy consumption for target classification in WSNs. Then, in Section 4, an experimental evaluation of our proposed approach is presented. The related works are discussed in Section 5. Finally, in Section 6, we conclude the paper with a brief summary and discussion of the findings presented.

2. Background

A rules set is often regarded as a good way of representing information or bits of knowledge [6], and rule based classifiers are very popular in various machine learning applications due to their easy usage for application. A rule based classifier consists of a set of IF-THEN rules, and each has a conjunction of attribute-value pairs (which will be called feature-value pair in the current paper) in the antecedent of the rule, followed by a class label in the rule consequent. A rule is illustrated as follows:

\[ R1: \text{IF weight = heavy \ AND \ radar = assembly} \]

THEN \text{armed Vehicles = yes,} \tag{1}

where the “IF” part is the rule antecedent, which consists of one or more feature-value pairs connected by the logical conjunction “AND.” And the “THEN” part is the rule consequent.

Given a target instance to be classified, if the condition of the rule antecedent (i.e., all the feature-value test) holds true, then it is said that the rule antecedent is satisfied. This also means that the instance is covered by that rule. A rule \( R \) is often evaluated by its coverage and accuracy. Given a dataset \( D \) with class labels, suppose \( n_{\text{covers}} \) is the number of instances covered by the rule \( R \), \( n_{\text{correct}} \) is the number of instances classified correctly, and \(|D|\) is the total number of instances in the dataset \( D \). The coverage of the rule \( R \) is defined following [6] in (2) as follows:

\[ \text{coverage}(R) = \frac{n_{\text{covers}}}{|D|}. \tag{2} \]

Thus, the accuracy of the rule \( R \) is defined in (3) as follows:

\[ \text{accuracy}(R) = \frac{n_{\text{correct}}}{n_{\text{covers}}}. \tag{3} \]

This means that a rule’s coverage is the percentage of instances that are covered by the rule, while for a rule’s accuracy, it is the instances covered by the rule, and see what percentage of them the rule can correctly classify. It is often easier to predict a given instance \( X \)’s class label using rule based classifier than the others (e.g., k-NN classifier). When a rule is satisfied by an instance \( X \), it is also called that the rule is triggered. For example, if \( X = (\text{weight = heavy}, \text{speed = medium}, \text{and radar = assembly}) \), then the rule \( R1 \) is triggered by \( X \).

Generating classification rules set via the intermediate form of decision tree is a popular technique in the research area of machine learning. This kind of classification algorithms is called the top-down induction of decision trees (TDIDT), also known as the “divide and conquer” approach. However, as pointed out by Cendrowska [7], the decision tree representation is itself the major cause of overfitting to the training data, and it also suffers from the replicated subtree problem [8]. An alternative approach is the “separate and conquer,” which is to take each class in turn and seek a way of covering all instances in it, at the same time excluding instances not in the class. This is called a covering algorithm (which is also known as the modular classification inductive learning algorithm). The modular classification inductive learning algorithms have their several modern representatives, including CN2 [9], PIPPER [10] and etc. Its most notable member is the Prism family of algorithms [11, 12], which is a direct competitor to the induction of decision trees.

The basic Prism algorithm [8] induces modular classification rules directly from the training set, in which the feature-value pairs are regarded as the discrete message in information theory. According to the basic principle of information theory, the amount of information about an event in message \( i \) is defined in (4) as follows:

\[ I(i) = \log_2 \left( \frac{\text{probability of event after receiving the message}}{\text{probability of event before receiving the message}} \right) \text{bits.} \tag{4} \]

The fundamental task of Prism algorithms is to find the feature-value pair \( f_x \), which contributes to most of the information about a specific classification \( y_i \), that is, maximizing \( I(y_i | f_x) \), which is defined in (5) as follows:

\[ I(y_i | f_x) = \log_2 \left( \frac{p(y_i | f_x)}{p(y_i)} \right) \text{bits.} \tag{5} \]
For each class \( y_l \) in turn, start with the complete training set \( D \) each time, and carry out the following steps:

**Step 1:** calculate the probability that class = \( y_l \), \( p(y_l | f_x) \), for each feature-value pair \( f_x \).  
**Step 2:** select the pair with the largest probability and create a subset of the training set comprising all the instances with the selected feature-value combination. 
**Step 3:** repeat Steps 1 and 2 for the subset until a subset that only contains instances of class \( y_l \) is reached. 
**Step 4:** remove all the instances covered by this rule from the training set. 
**Step 5:** repeat Steps 1 to 4 until all the instances of class \( y_l \) have been removed.

**Pseudocode 1:** Pseudocode for the basic Prism algorithm.

Where \( p(y_l) \) is the same for all \( f_x \), so it is only needed to find \( f_x \) for which \( p(y_l | f_x) \) is maximum. 

In the basic Prism framework, which is shown in Pseudocode 1, the training set is restored to its original state each time around. Steps 1 to 4 iterate over the classes and generate rules for each class in turn. While creating rules for class \( y_l \), remove the instances from the set until there is none of that class left. Whenever a rule is created, it starts with an empty form, which covers all the instances, and then it is restricted by adding tests until it covers only instances of the desired class. At each stage, the most promising feature-value test will be chosen, that is, the one that maximizes the accuracy of the rule. Furthermore, the ties are broken by selecting the test with greatest coverage.

Just like any classification rule induction algorithms, pruning methods can also be used to prevent Prism algorithm from overfitting on the training set. We consider J-pruning, an efficient facility for information theoretic pruning of modular classification rules, in the current paper. J-pruning is based on J-measure [13], a measure for the information content of a rule. Given a rule of the form \( IF \ F = f \ THEN \ Y = y \), the information content of the rule measured in bits of information is called J-measure, which is denoted by \( J(Y; F = f) \) as follows:

\[
J(Y; F = f) = p(f) \cdot j(Y; F = f), \quad (6)
\]

where \( p(f) \) is the probability that the antecedent of the rule will occur and \( j(F; F = f) \) is called cross entropy, which is defined in (7) as follows:

\[
j(Y; F = f) = p(y | f) \cdot \log_2 \left( \frac{p(y | f)}{p(y)} \right) + (1 - p(y | f)) \cdot \log_2 \left( \frac{1 - p(y | f)}{1 - p(y)} \right), \quad (7)
\]

where \( p(y) \) is the probability of the rule consequent (the probability that the consequent of the rule will be satisfied if we have no other information) and \( p(y | f) \) is the probability that the consequent of the rule will be satisfied if we know that the antecedent is satisfied.

This is how the J-pruning works, and it is easy to incorporate the pruning method to all versions of Prism algorithms. As mentioned before, in the basic Prism algorithm framework, due to different selection of target class, two other rule-generating strategies have been devised, which are called PrismTC and PrismTCS [14]. In order to have a comprehensive empirical study, we use three different algorithms in Section 5, which incorporate J-pruning into the three versions of Prism algorithms, called J-Prism, J-PrismTC, and J-PrismTCS.

### 3. Feature Selection Approach Based on Modular Rules

This section is used to introduce the particular feature selection algorithm for target classification in the wireless sensor networks context considered in this paper. Suppose a target instance \( X \) is represented by an \( N \)-dimensional feature vector; that is, \( X = (x_1, x_2, \ldots, x_N) \), where \( x_i \) is the value of the \( i \)th feature and \( i = 1, 2, \ldots, N \). As each feature is measured by exactly one sensor, which is assumed in Section 1, let \( S = \{s_1, s_2, \ldots, s_N\} \) be the set of available sensors and \( f(s_i) = x_i \). Thus, the maximal set of features that can possibly be used by the classifier is \( \{x_1, x_2, \ldots, x_N\} \). We just try to accomplish the target classification task using the number of the features as small as possible while maintaining a certain given classification accuracy.

The basic structure of the algorithm for target classification proposed in this paper is given in Pseudocode 2. The algorithm runs on the base station of a WSN. This system, in principle, can use any modular rule based classifier as its base classifier; thus, we called it feature selection approach based on modular rule classifier (FSMRC). For the purpose of simplicity, we use the Prism family of algorithms described in Section 2 as the base classifier in the current paper. Prism algorithms are popular in machine learning for the reason that they can induce modular classification rules directly.

The algorithm starts with ranking the features according to their occurrences in the rule set in line 2. Then, the algorithm initializes the set \( FV \) of feature-value pair to be used by the classifier to the empty set and the total energy consumption of the WSN to be 0 units (in our paper each sensor is assumed to consume one unit of energy for one measurement) (line 3). Next, it sequentially activates the sensors in the WSN, to find the feature-value pairs that appear in the antecedent of rules in the classifier, and the energy consumption will add one unit for each sensor’s measurement (lines 4 to 11). In line 12, it is checked whether there is one rule that is triggered by the feature-value pairs in \( FV \).
4 International Journal of Distributed Sensor Networks

If it is the case, the classification result is obtained by the Prism classifier (line 13). Otherwise, if it is not the case, the sensors for the other features in the antecedent of the rules that have subcondition for FV are activated. At the same time, the energy consumption will increase by one unit for each sensor’s measurement, where we assume that the total number of sensors to be activated is $k$. Hence, the classification result is achieved finally by the Prism classifier (line 14). Note that a Prism classifier is particularly suitable for such a rule based feature selection approach in this context. Because Prism always chooses a feature-value pair to maximize the probability of the desired classification, only several key feature-value pairs are enough to classify a new instance according to the rules in Prism classifier.

4. Experimental Evaluation

In this section, the proposed approach described before will be implemented and experimentally evaluated. Due to the lack of classification benchmarks that have been defined for the datasets used in the field of wireless sensor networks, we decide to use datasets from the UCI machine learning repository [15]. The experiments reported in this paper are conducted on nine datasets, and their basic information is shown in Table 1. All the datasets pose classification problems with only nominal features and no missing values. These conditions are suitable for the Prism family of algorithms to be used in a straightforward way. In addition these datasets have a rather large number of instances which makes them suitable to test the approach proposed in the current paper. In our experiments, each dataset is divided into a training set holding 70% of the instances and a test set holding the remaining 30% of the instances.

As mentioned in Section 1, in order to keep the energy consumption of sensors as low as possible, we investigate the energy consumption in each experiment of nine datasets. Instead of reporting the total number of energy consumption units directly, we give the efficiency for convenience, which is defined in (8) as follows:

$$
\text{energy\_efficiency} = 1 - \frac{\text{energy\_consumption}}{\text{total\_energy}},
$$

where $\text{energy\_consumption}$ and $\text{total\_energy}$ denote the units of energy actually consumed and total units of energy, respectively.

We assume that the test set to be classified consists of $M$ instances, and each feature value can only be measured exactly $N$ times before the battery of its sensor is exhausted. Thus, the total units of energy of all the sensors in such networks are $M * N$. This means that using all-features approach mentioned in Section 1, which uses the full set of $N$ features for each target to be classified, the total units of energy consumption by all the sensors are $M * N$. By contrast, in our selective approach, not all sensors are needed to acquire the corresponding feature values in the classification for each target.

We first investigate the accuracy in our system. Table 2 shows the comparison of all-features approach and FSMRC on accuracy, in which the accuracy of all-features approach is achieved using all the sensors in a WSN for each target classification.

What can be seen from Table 2 is that, in all nine datasets, our FSMRC approach performs either with the same accuracy or a little worse than all-features approach with the same base classifier. In fact, three times it produced the same accuracy as all-features approach. Taking a closer look on each base classifier, FSMRC using J-Prism is closest to all-features approach.

From the experiment results showed in Table 2, one can conclude that FSMRC can almost perform as good as all-features approach. More importantly, as energy consumption is the key issue of wireless sensor networks, in the experiments,
Table 1: Dataset and some of their characteristic properties.

| Dataset          | Number of instances | Number of features | Number of classes | Number of training instances | Number of test instances |
|------------------|---------------------|--------------------|-------------------|------------------------------|--------------------------|
| Congress vote    | 435                 | 16                 | 2                 | 300                          | 135                      |
| Car evaluation   | 1728                | 6                  | 4                 | 1210                         | 518                      |
| Chess            | 3196                | 36                 | 2                 | 2131                         | 1065                     |
| Connect-4        | 67557               | 42                 | 3                 | 45038                        | 22519                    |
| Molecular biology| 3190                | 61                 | 3                 | 2126                         | 1064                     |
| Nursery          | 12960               | 8                  | 5                 | 8640                         | 4320                     |
| Solar Flare      | 1389                | 10                 | 3                 | 926                          | 463                      |
| Balance Scale    | 625                 | 4                  | 3                 | 417                          | 208                      |
| Tic-Tac-Toe      | 958                 | 9                  | 2                 | 638                          | 320                      |

Table 2: Comparison of all-features approach and FSMRC on accuracy (%).

| Dataset          | J-Prism All-features | FSMRC All-features | J-PrismTC All-features | FSMRC All-features | J-PrismTCS All-features | FSMRC All-features |
|------------------|----------------------|--------------------|------------------------|--------------------|------------------------|--------------------|
| Congress vote    | 79.12                | 75.31              | 78.34                  | 71.56              | 84.15                  | 77.23              |
| Car evaluation   | 93.28                | 91.17              | 95.87                  | 89.15              | 97.12                  | 90.53              |
| Chess            | 95.37                | 89.23              | 97.13                  | 90.78              | 93.49                  | 88.24              |
| Connect-4        | 82.95                | 82.95              | 83.50                  | 81.53              | 87.56                  | 83.62              |
| Molecular biology| 79.35                | 79.35              | 80.26                  | 78.23              | 83.25                  | 80.12              |
| Nursery          | 83.17                | 82.95              | 88.31                  | 82.46              | 87.10                  | 80.65              |
| Solar Flare      | 71.65                | 70.31              | 72.89                  | 69.94              | 78.32                  | 75.53              |
| Balance Scale    | 89.80                | 89.80              | 90.41                  | 83.25              | 92.25                  | 85.28              |
| Tic-Tac-Toe      | 90.85                | 89.15              | 91.63                  | 84.71              | 93.46                  | 87.12              |

we pay more attention to investigating the energy efficiency of our system, which has been defined in (8). A representation of the energy consumption results appears in Figure 1.

What can be seen in Figure 1 is that in all cases the proposed system is energy efficient. Considering the different selection of base classifiers, one may also conclude that the choice of a proper rule based algorithm as the base classifier is not a critical issue. For the nine datasets used for evaluation in our system, in most cases, the energy efficiency is largely between 20% and 30%, which means that the energy consumption can be reduced by nearly one-third, comparing to the all-features approach. Clearly, what can be concluded from the above experimental results is that our system is noticeably energy efficient without losing much classification accuracy in a wireless sensor network with sensitive data.

5. Related Works

The areas of work related to our approach are target classification in energy-constrained wireless sensor networks and rule based classifiers combination. Several approaches to minimizing energy consumption for target classification in WSN have been presented in the literature. Wang and Xiao [16] provide an in-depth analysis on the mechanism for the energy-efficient sensors scheduling in WSN. Krause et al. [17] investigate the problem of monitoring spatial phenomena using wireless sensors with restricted battery capacity. The main concern is to decide where to locate the sensors to best predict the phenomenon at the locations which cannot be sensed. Their focus is on locating the sensors rather than on selecting features corresponding to different sensors. Tian and Georganas [18] present a node-scheduling scheme for reducing the overall energy consumption of the system. Their main approach is turning off some redundant nodes. However, FSMRC runs in a WSN where the sensor nodes cannot be turned off.

Concerning rule based classifiers combination, the idea that modular rules can outperform decision tree by achieving higher classification accuracy was first discussed in [7]. Since modular rules are easy to be carried out in the task of classification, various approaches have been proposed to combine classifiers (which is also known as ensemble classifiers). More recently, Stahl and Bramer [19] present and evaluate ensemble rule classifiers based on the "separate and conquer" approach, called random Prism. The main idea is inspired by Random Forest [20]. These approaches focus on improving the classifier’s accuracy, with no consideration for any other cost function. In addition, these rule classifiers combinations are fundamentally parallel architecture, which means a given target instance is processed in parallel, while FSMRC is a serial or sequential approach, characterized by activating sensors in sequential order.

6. Conclusions

In this paper, a rule based feature selection approach for target classification is proposed to minimize the energy consumption in wireless sensor networks. Our research is
motivated by sensitive data applications, in which it is critical to prevent the sensitive original historical data from leaking. This means that it is better to run rule based classifiers rather than instance based classifiers. It is well known that rules are a good way of representing information or bits of knowledge, and rule based classifiers are very popular in various machine learning applications due to their easy usage for application.

Our proposed approach can be applied in conjunction with any known rule based classifier. In this paper, we choose Prism classifier, because its maximizing information gain procedure is particularly suitable for such a feature selection approach in this research context. With the Prism family of algorithms as our system’s base classifier, we devise a specific algorithm for target classification, which is conducted on the base station in wireless sensor networks.

The proposed approach was evaluated by experiments on nine datasets from the UCI machine learning repository. The results of the experiments revealed that our system is noticeably energy efficient without losing much classification accuracy in wireless sensor networks. In fact, the energy efficiency can be improved by nearly one-third, comparing to all-features approach. That is, we can achieve equivalent accuracy to all-features approach for more target classification tasks in energy-constrained wireless sensor networks.

Ongoing work on FSMRC comprises investigating rule based classifiers other than the Prism family of algorithms. Furthermore, an extension of the experiments to the datasets obtained from real wireless sensor networks would be desirable.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This research is supported by the National Natural Science Foundation of China (Grant no. 71371184) and the Specialized Research Fund for the Doctoral Program of Higher Education of China (Grant no. 20124307110023). The authors would like to thank the anonymous reviewers and guest editors, who gave valuable suggestion that has helped to improve the quality of the paper.

References

[1] K. Bhaduri and M. Stolpe, Distributed Data Mining in Sensor Networks, Managing and Mining Sensor Data, Springer, New York, NY, USA, 2013.
[2] Y. Sun and H. Qi, “Dynamic target classification in wireless sensor networks,” in Proceedings of the 19th International Conference on Pattern Recognition (ICPR’08), December 2008.
[3] J. Csurik, P. Bertholet, and H. Bunke, “Sequential classifier combination for pattern recognition in wireless sensor networks,” Multiple Classifier Systems, pp. 187–196, Springer, Berlin, Germany, 2011.
[4] I. Guyon, Ed., Feature Extraction: Foundations and Applications, vol. 207, Springer, 2006.
[5] I. Guyon and A. Elisseeff, “An introduction to variable and feature selection,” The Journal of Machine Learning Research, vol. 3, pp. 1157–1182, 2003.
[6] J. Han, M. Kamber, and J. Pei, Data Mining: Concepts and Techniques, Morgan Kaufmann, 2006.
[7] J. Cendrowska, “PRISM: an algorithm for inducing modular rules,” International Journal of Man-Machine Studies, vol. 27, no. 4, pp. 349–370, 1988.
[8] I. H. Witten, E. Frank, and M. A. Hall, Data Mining: Practical Machine Learning Tools and Techniques, Elsevier, 2011.
[9] P. Clark and T. Niblett, “The CN2 induction algorithm,” Machine Learning, vol. 3, no. 4, pp. 261–283, 1989.
[10] W. W. Cohen, “Fast effective rule induction,” in Proceedings of the 12th International Conference on Machine Learning (ICML’95), vol. 95, 1995.
[11] M. Bramer, “Inducer: a public domain workbench for data mining,” International Journal of Systems Science, vol. 36, no. 14, pp. 909–919, 2005.
[12] F. Stahl and M. Bramer, “Jmax-pruning: a facility for the information theoretic pruning of modular classification rules,” Knowledge-Based Systems, vol. 29, pp. 12–19, 2012.
[13] P. Smyth and R. M. Goodman, “An information theoretic approach to rule induction from databases,” IEEE Transactions on Knowledge and Data Engineering, vol. 4, no. 4, pp. 301–316, 1992.
[14] M. Bramer, An Information-Theoretic Approach to the Pre-Pruning of Classification Rules, Intelligent Information processing, Springer, New York, NY, USA, 2002.
[15] A. Frank and A. Asuncion, “UCI machine learning repository,” 15:22, 2011, http://archive.ics.uci.edu/ml.
[16] L. Wang and Y. Xiao, “A survey of energy-efficient scheduling mechanisms in sensor networks,” Mobile Networks and Applications, vol. 11, no. 5, pp. 723–740, 2006.
[17] A. Krause, R. Rajagopal, A. Gupta, and C. Guestrin, “Simultaneous placement and scheduling of sensors,” in Proceedings of the International Conference on Information Processing in Sensor Networks (IPSN’09), pp. 181–192, April 2009.
[18] D. Tian and N. D. Georganas, “A coverage-preserving node scheduling scheme for large wireless sensor networks,” in *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications*, pp. 32–41, ACM, September 2002.

[19] F. Stahl and M. Bramer, “Random Prism: a noise-tolerant alternative to Random Forests,” *Expert Systems*, 2013.

[20] L. Breiman, “Random forests,” *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.
Submit your manuscripts at
http://www.hindawi.com