

Towards Continuous Monitoring of Environment under Uncertainty: A Fuzzy Granular Decision Tree Approach

Preetham N. Reddy
Department of EEE
BITS Pilani, Goa Campus
Goa 403726, India
f2015174@goa.bits-pilani.ac.in

Sahith N. Dambekodi
Department of EEE
BITS Pilani, Goa Campus
Goa 403726, India
f2015192@goa.bits-pilani.ac.in

Tirtharaj Dash
Data Science Research Group
BITS Pilani, Goa Campus
Goa 403726, India
tirtharaj@goa.bits-pilani.ac.in

ABSTRACT

Smart monitoring of environment has been an essential area of research where decision-making process is inevitable. Reliability of the whole system depends on the stability and consistency of its decision-making unit. Real-time decision making is another challenge in the field on which the research community has been focusing on improving the performance of the underlying models. The underlying models are usually the learning models, that act as a smart engine after being sufficiently trained for the process. In this paper, we propose to use a decision tree model that has the capability of handling uncertainty in the acquired data from the environment. The resulting model is called as Fuzzy Granular Decision Tree (FGDT). Series of evaluation of FGDT shows that the model is stable and powerful for the presently considered problem.

CCS CONCEPTS

•Computing methodologies →Machine learning approaches; Classification and regression trees; Uncertainty quantification;

KEYWORDS

Environment Monitoring; Decision Tree; Fuzzy Logic; Gas Sensors

ACM Reference format:

Preetham N. Reddy, Sahith N. Dambekodi, and Tirtharaj Dash. 2017. Towards Continuous Monitoring of Environment under Uncertainty: A Fuzzy Granular Decision Tree Approach. In *Proceedings of Workshop on Development aspects of Intelligent Adaptive Systems, Co-located with 10th ISEC-2017, Jaipur, India, (DIAS 2017)*, 5 pages. DOI:

1 INTRODUCTION

Smart monitoring of environment is increasingly becoming popular because of the obvious fact that very little human intervention is required for such systems to perform. Moreover, it has been a very challenging and yet interesting area of research in the last several years [1]. This field has motivated the research community to design automated and intelligent models (or systems) towards continuous monitoring of an environment in industrial plants, medical environment or biological processes. However, designing an *almost-accurate* system has been a challenge given the background

complexity of the environment or availability of the required equipment for the process. In the past, this has led the theoretical and computational computer scientists to develop mathematical models that could almost simulate the actual target environment.

Towards advancing the field forward, a research was carried out very recently by Huerta et al.[2] that focused on online decorrelation of humidity and temperature in chemical sensors for continuous monitoring. The work focused on automated processing of data from simultaneous and continuous readings of the variations in humidity and temperature in the targeted environment. Their work included eight(8) metal-oxide sensors (MOX) sensors that were continuously sensing the environment for 537 days with a sampling rate of 1 sample per second [2]. They estimated the effects of variations in air humidity and temperature on the chemical sensors signals by using standard energy band model with an assumption that the variations in sensor conductivity can be expressed as a nonlinear function of changes in the energy bands in the presence of external humidity and temperature. Their study showed that the major factors that were affecting the environment were the changes in humidity and the correlated changes in temperature and humidity. To visualize the process, they used a gas discrimination system that could discriminate among banana, wine, and baseline response. They had used a variant of Support Vector Machine (SVM) algorithm to build the discriminatory model for the process.

In the process of continuous environment monitoring, there lies a certain degree of uncertainty in the acquired data. An uncertainty in the acquired data could lead to the failure or adverse functioning of the system. The uncertainty could arise due to the following situations such as failures of sensors, a sudden change in environment due to additional and unknown factors. In such a case, the discriminatory model that was built may not be trustworthy and hence the results should, in turn, be imprecise. This impreciseness or degree of uncertainty in the acquired data or desired outcome is called ‘fuzziness’. An adaptive system that is meant to serve for the purpose of continuous environment monitoring should not only just adapt to the environmental changes with time but also should be robust with respect to the uncertainties as read through its sensors. To solve the purpose of handling uncertainty or the fuzziness present in the acquired data, and to make a robust decision based on the data, we propose a granular method – that specifically captures the uncertain inputs – to be induced within an adaptive system. The research carried out by [2] was experimentally sound and our present work mainly serves the purpose of embedding the uncertainty aspect of the work. In our present work, we redesign a discriminatory model that also incorporates possible uncertainty in the acquired data. It should be noted that our work is validated

with the experimental data from [2], further information of which has been presented in the following sections.

The rest of the paper is organized as follows. The proposed and implemented model has been discussed in the section 2. The results that have been obtained for the present problem has been elaborated in section 3. The obtained results are discussed in the section 4 and the paper has been concluded in the section 5.

2 METHODOLOGY

Our proposed work has following major phases: data acquisition phase, uncertainty handling phase, discrimination phase.

2.1 Data acquisition phase

In the data acquisition phase, the sensors acquire the data from the environment for further processing in later phases. For more information, one could refer [2].

2.2 Handling uncertainty in acquired data

In this work, uncertainty in the acquired data has been taken care with the use of feature transformation using a special fuzzy membership function [3, 4]. The membership function transforms a single sensor reading (datum) to granules containing the information of different belongingness. For example, a temperature reading of 38° can be called as a low temperature, medium temperature or high temperature. Hence, it can have some degree of belongingness to each of low, medium and high temperature. In this work, we chose to use the Gaussian fuzzy membership function that is given in equation (1) where a is the maximum value of the membership function, σ is the standard deviation of the readings obtained by a sensor. The value of \hat{x} is different for each of the low, medium and high belongingness. For transformation to low degree of belongingness, $\hat{x} = \min\{\mathbf{x}\}$; for transformation to medium degree of belongingness $\hat{x} = \text{mean}\{\mathbf{x}\}$; and for transformation to higher degree of belongingness $\hat{x} = \max\{\mathbf{x}\}$. Please note that \mathbf{x} is a vector that contains all the sensor readings for any particular feature. For i th feature (alternatively, for i th sensor), \mathbf{x} should be represented as x_i .

$$\mu(x) = a \exp\left(-\frac{(x - \hat{x})^2}{2\sigma^2}\right) \quad (1)$$

The full transformation can be visualized in Figure 1 that represents the curve for low, medium and high membership function (from left to right). The belongingness can have a maximum value of 1 ($a = 1$) and a minimum value of 0. So, for a single datum, the transformation function generates three different granules. At a particular time t , if there are n sensors, there can be n sensor readings, called features. After the transformation, the feature vector gets transformed to a higher dimension of $3n$.

A major intuition behind such a high dimensional transformation is that a discriminatory model is expected to perform better with high dimension than in low dimension. This is as per Cover's theorem on the separability of patterns [5]. Since this work is focused on discrimination of the sensor readings into different classes, the transformation should make the patterns separable in higher dimensional space, if they are not easily separable in a lower dimension. In a further section, we shall see that this is indeed the case for this specific problem.

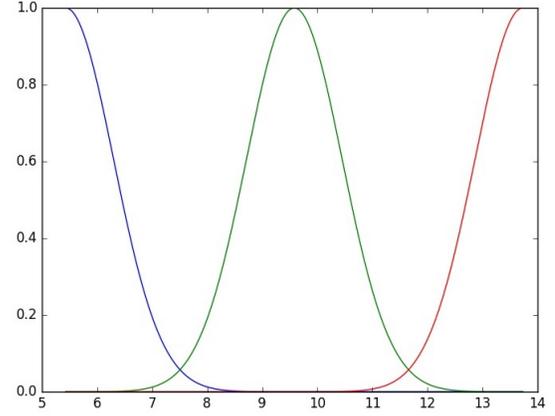


Figure 1: A sample Gaussian fuzzy membership transformation function (X-axis: value of a sample sensor reading, Y-axis: degree of belongingness, μ)

2.3 Discrimination phase

Discrimination is the process of assigning a class level to a set of inputs i.e. a set of sensor readings at a particular time. Consider a set of sensor readings at any particular time be represented as $\{x_1, x_2, \dots, x_n\}$ whose class is unknown. It should be noted that there are three classes in the present work: banana, wine, and baseline response. After the fuzzy transformation, the vector can be represented as

$$\mathbf{x} = \{x_1^{low}, x_1^{medium}, x_1^{high}, x_2^{low}, x_2^{medium}, x_2^{high}, \dots, x_n^{low}, x_n^{medium}, x_n^{high}\} \quad (2)$$

The discrimination phase works with a model that has been built with prior knowledge, called training data, to discriminate the sensor readings in the real time, called test data. In this work, we implement a decision tree (DT) classifier that works with the transformed fuzzy granular feature space that has been obtained from the second phase. Therefore, the whole model can be known as Fuzzy Granular Decision Tree (FGDT). The principal reason behind choosing DT over other machine learning models is that it is non-parametric that can learn from the training data in a supervised manner. More specifically, we implement CART that is very similar to C4.5 decision tree [6]. The FGDT predicts the class of a set of sensor readings by learning simple decision rules inferred from the features. For completeness and clarity of the readers, we present the working principle of the FGDT as follows.

Given a training dataset in fuzzy granular input feature space where each of the features is represented as $x_i \in R^{3n}$. Note that each x_i is a vector in $3n$ dimension represented as equation (2). Let the class labels be represented as d . The FGDT recursively partitions the training pattern space such that the patterns with same class labels are grouped together. The following Algorithm 1 explains the generic steps of the FGDT that take the fuzzy granular sensor readings training instances ($\tilde{\mathcal{D}}_{train}$) and generates a decision tree from that ($\tilde{\mathcal{T}}$).¹

¹The $\tilde{\cdot}$ symbol is used to represent uncertainty in input data and the tree is generated from this fuzzy granular data.

Data: A transformed training set, $\tilde{\mathcal{D}}_{train}$
Result: A decision tree, \tilde{T}
if All the instances belong to same class **then**
 | Return a \tilde{T} with a leaf node labelled with the class;
end
if $\tilde{\mathcal{D}}_{train} = \{\phi\}$ **then**
 | Return $\tilde{T} = \phi$ with warning;
end
if If there is no feature in $\tilde{\mathcal{D}}_{train}$ **then**
 | Returns a leaf labeled with the most frequent class or the
 | disjunction of all the classes)
end
while Until stopping condition is not met **do**
 | Find the feature $j \in \tilde{\mathcal{D}}_{train}$ with highest informational
 | gain or lowest impurity;
 | Based on feature j , Split the present node of \tilde{T} to form
 | sub-trees $\tilde{T}_{left}, \tilde{T}_{right}$;
 | Repeat for $\tilde{T}_{left}, \tilde{T}_{right}$;
end
Algorithm 1: FGDT algorithm (The first three **if** statements are called base cases. An impurity function can also be used for the information gain: lower the impurity, higher the information gain and vice versa.)

The stopping conditions while generating the \tilde{T} from $\tilde{\mathcal{D}}_{train}$ could be one of the following. A fully grown most generalized tree has been obtained, the training pattern space has been well partitioned into multiple sub-spaces based on classes, the generated tree returns lowest error on validation data². For measuring the impurity, an ‘entropy’ estimate is used which is given in equation (3).

$$H(\mathbf{X}, \tilde{T}) = - \sum_{i=1}^m P(\mathbf{X}_i) \log_2(P(\mathbf{X}_i)) \quad (3)$$

Here, $P(\cdot)$ is the probability estimate. Based on the estimation in equation (3), the entropy before split and entropy after splitting a node is computed for each feature. The feature which would provide lowest entropy after split is considered to be split up based on some condition, usually a threshold. Entropy after split is computed from the entropy estimate of each of the new sub-trees as

$$H_{afterSplit}(\mathbf{X}, \tilde{T}) = - \sum_{i=1}^k P(\mathbf{X}) \log_2(H_i(\mathbf{X}_i), \tilde{T}_i), \quad (4)$$

where, k is number of splits at a node of the tree, $H_i(\mathbf{X}_i), \tilde{T}_i$ is the entropy measure for the sub-tree that would be produced after split. By using these entropy estimates that is entropy before split and entropy after split, the information gain which is denoted as G can be computed as

$$G = H_{beforeSplit}(\mathbf{X}, \tilde{T}) - H_{afterSplit}(\mathbf{X}, \tilde{T}) \quad (5)$$

The feature that would produce the highest information gain should be used to split at a particular node in the tree \tilde{T} . It should be noted that the discussion and algorithm implemented in this work splits a

node of the tree into two sub-trees (binary split). It is also possible to split a node into multiple sub-trees (multiple splits) based on more than one thresholds at the node [6].

Effect on computational complexity? – It is quite obvious that the size of the feature set is increased by the transformation into the fuzzy feature space under the three membership functions such as low, medium and high. If we consider that there are n_{input} number of features in the original data, after the transformation into the fuzzy pattern space, the size becomes $3 \times n_{input}$. If the time complexity for the classical decision tree learning is $\mathcal{T}(n_{input})$, the FGDT has a time complexity of $\mathcal{T}(3 \times n_{input})$. However, it should be noted that a fuzzy granular decision tree learner would have to learn once with the available training data; and the learned model is just deployed for its testing. So, this cost over time would occur just once. Moreover, with the availability of high-performance computing architectures, this complexity could be lowered and scalability of the proposed model could be improved. Moreover, although a decision tree learner learns from the data by partitioning the space into multiple subspaces with conditions, it is already shown that the decision tree scales well with higher dimensional data [7].

3 RESULTS

An adaptive system should not only just adapt to the environmental changes but also should be robust with respect to the uncertainties as reading through the sensors. This could be done via robust modeling of the environment or via robust decision making withing the system. However, our present work demonstrates the former category where the implemented model adapts to the changes in the environment by also incorporating the possible uncertainties by transforming the problem into an imprecise problem. To support our claim about the improvement, we used the experimental data available from the work of Huerta et al. [2] to study the performance of our proposed fuzzy granular model, FGDT. To elaborate, to evaluate the impact on discrimination performance of our proposed FGDT due to decorrelation of signals from temperature and humidity sensors, four different feature sets were used. The first set of features is a set of raw sensor time series (RS); the second feature set is a set of raw sensor data with humidity and temperature (RS, T, H); the third set of features is a set of filtered data (FS) by decorrelating sensors; the fourth set of features is a set of raw sensor data with filtered sensor data (RS, FS). We used these same feature set for a proper evaluation of our proposed model. However, additionally, we also use a new dataset that also contains ‘t0’ and ‘dt’ along with filtered sensor data (FS) for evaluation.

To properly estimate the generalization ability of the proposed FGDT model, we used the standard procedures in machine learning to evaluate the performance of FGDT when discriminating samples not used for training the classifier so as to remove evaluation bias during the testing phase. There are 919438 number of sensor-reading instances in the dataset. The available data was randomly partitioned into 80% and 20%. The partition with 20% of data was used for the independent test. The 80% partition was used for 5-fold cross validation (5CV). For a fair comparison, we used accuracy as the performance evaluation measure for this work as the same was also used in [2], although accuracy should not be always considered

²A set of instances that were not used during the training process but are used to check the performance of the generated model after training. These are not test data.

reliable [8]. However, as the number of instances in the dataset is very large, at the order of 9×10^5 , it would not have a large effect on the accuracy measure. We considered the fold-wise performance in 5CV along with independent test performance for the evaluation purpose. The results have been noted and depicted in Table 1. To have a better evaluation of the model, the 20% partition was not made fixed from the beginning, rather, with each fold evaluation, the test partition has been chosen.

A comparison of the result obtained by our proposed FGDT with the results obtained by Huerta et al.[2], as shown in Table 2, suggests that the proposed FGDT model is superior to the ISVM model for this problem for most of the datasets with selected features. However, for the dataset with only FS as feature set, the performance of FGDT is far lower than the ISVM model. This is probably because of the fact that the FS dataset is quite difficult to be separable which could be revealed from a sample plot between F1 vs F2 of the FS feature set (see Figure 2).

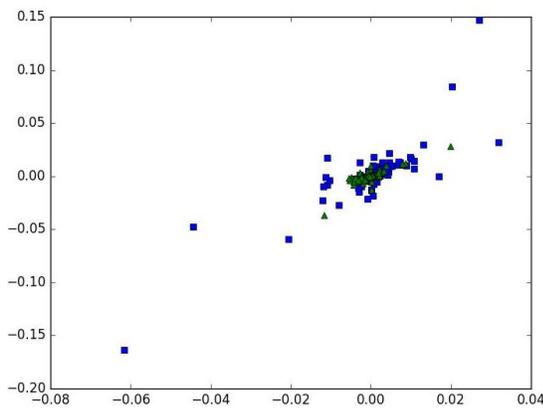


Figure 2: Scatter plot between F1 and F2

4 DISCUSSION

The FGDT algorithm does handle uncertainty in the acquired data by transforming it to a feature space that is higher in dimension than that of the original data. This could lead the decision tree classifier capture the underlying relationship between the inputs and outputs in a better way. This is evident from the fact that the higher dimension of the transformed data makes the visibility sparse and hence drawing a clear boundary between two different classes becomes easy [5], which could not have been possible by using SVM classifier as used in [2]. More specifically, when a lower dimensional input space is projected onto a higher dimensional space, the sparsity in the projected space increases and thereby increasing the chance of learning a possibly-optimal hyperplane that would be serving as the boundary among various resulting groups or classes of patterns. Technically, the distance between any pattern in the project space and the learned hyperplane does improve which makes a strong impact on the training of the decision tree learner. This interpretation could probably be true because of the fact that a more generalized model that is built after training and validation would possess a higher capability of generating accurate

results during independent real-time tests. Moreover, the number of instances in a training set and the number of features do also play crucial roles in the process. With a higher number of instances with a small set of features might not generate an adequate function that would generate output from the inputs. After transformation to fuzzy space and hence increasing the dimension of the input space, such a function would be possible for the purpose. This could lead to the proper generalization of the FGDT and hence the independent test performance. One should note that the present work uses the CART algorithm available in Scikit-learn [9]. However, one could try many other machine learning approaches for the same task after incorporating our proposed fuzzy granular transformation approach as a step in feature engineering before training of the learning model.

5 CONCLUSIONS

The uncertainty arising in the acquired data from the environment in the process of continuous home monitoring has been handled by developing a new fuzzy granular approach that has been combined with a decision tree for decision making. The proposed model FGDT has been evaluated with regard to experimentally validated data from continuous monitoring environment. The performance of our FGDT model is found to be superior to a recently published work on the same problem. Moreover, the statistical results also show that FGDT has not only better discrimination capability but also it is quite stable and consistent.

Although the present work discusses the applicability of fuzzy modeling towards the uncertainty handling in the aspect of environmental monitoring systems; there are numerous possible applications of such techniques in real-world problems such as medical diagnosis [10], robotic navigation control mechanisms [11, 12], handling uncertainty in software testing [13]. All these mentioned real-world problems may not always be subject to precise computation, and hence uncertainty handling methods such as fuzzy granular decision tree (FGDT) approach would aid to achieve greater performance of the intended adaptive systems.

REFERENCES

- [1] Joey F Boatman and Bryan S Reichel. Environment monitoring system, April 6 1999. US Patent 5,892,690.
- [2] Ramon Huerta, Thiago Mosqueiro, Jordi Fonollosa, Nikolai F Rulkov, and Irene Rodriguez-Lujan. Online decorrelation of humidity and temperature in chemical sensors for continuous monitoring. *Chemometrics and Intelligent Laboratory Systems*, 157:169–176, 2016.
- [3] Sushmita Mitra, Rajat K De, and Sankar K Pal. Knowledge-based fuzzy mlp for classification and rule generation. *IEEE Transactions on Neural Networks*, 8(6):1338–1350, 1997.
- [4] Tirtharaj Dash, Sanjib Kumar Nayak, and HS Behera. Hybrid gravitational search and particle swarm based fuzzy mlp for medical data classification. In *Computational Intelligence in Data Mining-Volume 1*, pages 35–43. Springer, 2015.
- [5] Thomas M Cover. Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition. *IEEE transactions on electronic computers*, (3):326–334, 1965.
- [6] Leo Breiman, Jerome Friedman, Charles J Stone, and Richard A Olshen. *Classification and regression trees*. CRC press, 1984.
- [7] Guosheng Lin, Chunhua Shen, Qinfeng Shi, Anton van den Hengel, and David Suter. Fast supervised hashing with decision trees for high-dimensional data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1963–1970, 2014.
- [8] Siddharth Dinesh and Tirtharaj Dash. Reliable evaluation of neural network for multiclass classification of real-world data. *arXiv preprint arXiv:1612.00671*, 2016.
- [9] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss,

Table 1: Performance (accuracy) of FGDT with different feature set for 5CV and independent test

Features→	RS		RS, T, H		FS		FS, RS		FS, t0, dt	
Fold	CV	Test	CV	Test	CV	Test	CV	Test	CV	Test
1	96.99	97.01	97.03	97.03	37.30	37.32	96.96	96.94	99.10	99.09
2	96.95	96.99	96.96	97.03	37.20	37.32	96.95	96.97	99.06	99.09
3	96.91	97.01	96.96	97.04	37.27	37.31	96.99	96.96	99.06	99.09
4	96.97	97.00	96.99	97.04	37.36	37.32	97.01	96.98	99.11	99.09
5	96.93	97.01	96.96	97.04	37.38	37.32	96.90	96.98	99.11	99.09
average	95.95	97.01	96.98	97.04	37.30	37.33	96.96	96.97	99.10	99.09
±std.dev.	±0.03	±0.01	±0.03	±0.01	±0.01	±0.01	±0.04	±0.01	±0.02	±0.01

Table 2: Comparison of FGDT with results obtained by Huerta et al. [2]

Feature Set	ISVM [2]		FGDT	
	CV	Test	CV	Test
RS	78.5	76.5	95.95 ± 0.03	97.01 ± 0.01
RS, T, H	73.3	71.1	96.98 ± 0.03	96.98 ± 0.03
FS	72.4	71.2	37.30 ± 0.01	37.33 ± 0.01
RS, FS	82.6	80.9	96.96 ± 0.04	96.97 ± 0.01
FS, t0, dt	-	-	96.96 ± 0.04	96.97 ± 0.01

Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(Oct):2825–2830, 2011.

- [10] Tanistha Nayak, Tirtharaj Dash, D Chandrasekhar Rao, and Prabhat K Sahu. Evolutionary neural networks versus adaptive resonance theory net for breast cancer diagnosis. In *Proceedings of the International Conference on Informatics and Analytics*, page 97. ACM, 2016.
- [11] Tirtharaj Dash, Tanistha Nayak, and Rakesh Ranjan Swain. Controlling wall following robot navigation based on gravitational search and feed forward neural

network. In *Proceedings of the 2nd international conference on perception and machine intelligence*, pages 196–200. ACM, 2015.

- [12] Tirtharaj Dash. Automatic navigation of wall following mobile robot using adaptive resonance theory of type-1. *Biologically Inspired Cognitive Architectures*, 12:1–8, 2015.
- [13] Salman Abdul Moiz. Uncertainty in software testing. In *Trends in Software Testing*, pages 67–87. Springer, 2017.