# A Study and Analysis on Cellular Automata based Classifier in Data Mining

Mayank Arya Chandra
IMSEC Ghaziabad, India

Vidushi
HMRITM, Delhi, India

## ABSTRACT

In the era of Information Technology, information flow has been enormously increased. Data mining techniques are widely used and accepted to retrieve information from various data. Cellular automata based techniques have been extensively reported in complex adaptive system. In this we present a survey of cellular automata as classifier.

## Keywords

Cellular Automata, Data Mining, Classifier.

## 1. INTRODUCTION

Cellular automata are discrete systems in which a population of cells evolves from generation to generation on the basis of local transitions rules [1][2].Cellular automata concept emerged out from self replicating system [3]. After that elementary cellular automata concept was published by Stephen Wolfram. Cellular automata do computation in a distributed manner. It works with a finite plane instead of infinite plane. We also refer cellular automata as a reversible system because the current status of a cell is exactly reversible to it previous stage.

A cellular automaton (CA) consists of a grid of cells, usually in one or two dimensions. Each cell takes on one of a set of finite, discrete values. Each cell has a finite and fixed set of neighbors, called its neighborhood. Various neighborhood [4] definitions have been used. Two common two-dimensional neighborhoods [5] are the von Neumann neighborhood [6], in which each cell has neighbors to the north, south, east and west; and the Moore neighborhood [7], which adds the diagonal cells to the northeast, southeast, southwest and northwest. In general, in a d-dimensional space, a cell's von Neumann neighborhood will contain 2D cells and its Moore neighborhood will contain $3D - 1$ cells. A grid is "seeded" with initial values, and then the CA progresses through a series of discrete timesteps.

At each timestep, called a generation, each cell computes its new contents by examining the cells in its immediate neighborhood. To these values it then applies its update rule to compute its new state. Each cell follows the same update rule [8] , and all cells' contents are updated simultaneously and synchronously. A critical characteristic of CAs is that the update rule examines only its neighboring cells so its processing is entirely local; no global or macro grid characteristics are computed.

Data Mining means the discovery or extraction of useful information in database [9][10][11]. Classification is the process of Data Mining. There are basically three models in which classification [12][13] can be represented. These are: - If-then rules, A decision tree, Neural Network.

## 2. GENERAL ISSUES IN CLASSIFIER DESIGN

### 2.1 DECISION TREE CLASSIFIER

Decision tree classifies visualize what steps have to be taken to achieve classification. Through decision tree classifiers [14][15] we organize dataset. It is a very common classification scheme. Decision tree always begins with a root node, also called as the parent node of all nodes. This classifier uses a routing technique [16][17] from root node to the leaf node of the dataset. It handles the dataset in a canonical form. The simplest data for these types of classifiers [18] is numeric data. This classifiers works recursively. This is generally used in data mining. Complexity reduction and automatic feature selection can be done by decision tree classifiers.

### 2.2 NAÏVE BAVES CLASSIFIER

It uses many quality measures as well as search algorithms. This classifier undertakes that any particular feature presence of a class is not related to the other feature presence, given the class variable [19]. For instance, vegetable Aubergines if it is black, long and about 6'' in length. Naive Bayes classifier will take all of these properties to independently contribute to the probability that this vegetable is an Aubergines, even if these features depends on each other

### 2.3 RADIAL BASIS FUNCTION CLASSIFIER

This is also a network based classifier. This classifier is capable to classify data with more objects in one go [20][21][22]. It consists three layers. First is Input layer second is hidden layer and the third one is output layer. It has many advantages over SVM (support vector machine) [23][24][25]. Because SVM takes time to select model and it doesn't perform where new instances are continuously added in already huge database. The main thing with RBF classifier is that the boundary instances contain more crucial information in respect of the inner parts instances.

### 2.4 LAZY CLASSIFIERS

Lazy classifiers have lazy learning method [26][27][28] approach. These methods doesn't do generalization and doesn't build the classify model until a query is provided as input. This classifier use CAR (Class association rules) for testing the class instances. It does a global search for satisfying the rules. This non-eager classifier focus on the test instances and it increase the chance to generate additional rules and it is very useful to classify the test instances.

## 2.5 LINEAR CLASSIFICATION

In data mining and machine learning linear classification is useful tool. Linear Classifiers works directly on original data input space. An important goal of the recent research on linear classification [29][30][31] is to develop fast optimization algorithms [32]for training

## 2.6 XCS CLASSIFIERS

In 1995, Wilson [33] [34] introduced this classifier . XCS classifier is a classifier system that retains important aspects of Holland's model [35]and it also improve this model in some aspect [36][37][38]. It is an adaptive classifier [39]. It has ability to choose the appropriate action. Every time a set of action is produced and a sure chance to create a genetic algorithm that will operate on set of instances.

## 2.7 Cellular Automata History

| Scientist | Year | Work |
|---|---|---|
| Von Neumann | 1950's | first introduced the concept of cellular automata[40][1] |
| Stan Ulam | 1950's | Concept of cellular automata is also put forward by Ulam[40][1] |
| John Conway | 1970 | Game of life[40][1][41] |
| Stephen Wolfram | In the beginning of the eighties | one-dimensional cellular automata rules or Wolfram rules are studied and showed by him. |

***Table 1***

## 3. CELLULAR AUTOMATA IN OTHER AREA OF COMPUTING

## 3.1 BIOLOGICAL

Biological systems are one of the complex system examples. Neural network [42][43][44] is a part of biological system [45][46] [47] domain. Human brain has a complicated architecture. It is built of large number of neurons that interact with each other mutually. Brain engineering [48][49]is the latest attempt to understand the brain-architecture. The base of Brain engineering system is cellular automata, that manage the growth and communication between neurons. Bayesian theory [27][50] is one of the approach that we follow in biological systems. It is based on statistical classification. We can combine the output of multiple networks for decision-making in higher-level systems

## 3.2 ECONOMIC

Economic systems are complicated systems. It is very tough to implement it practically. But theoretically it is easy to understand through cellular automata. But still it is too far to real applications. Complex systems are understandable in high dimension but mathematically it is hard to implement it through cellular automata. There are very few research of cellular automata in non-linear economics [51][52]. Complex economic systems have a very similar architecture to cellular automata. Its features should be used in non-linear economic system.

## 3.3 GAME OF LIFE

Different games can be developed with the help of cellular automata. Game of life [41][2][46]is the example of CA game. It is zero player game, which means initial stage determines its evolution. It has two-dimensional infinite grid of square cells. A cell can interact with its 8 neighborhood cells (B3/S23). The pattern may be static or repeating. In starting game of life, a cell has 2 to 3 neighbors after that 6 neighbors (B6/S16). After that 2D and 3D variations came into the picture. There are many programs are available online based on life.

## 3.4 PARALLEL COMPUTING

Cellular automata is also used to build parallel multipliers [52]. In image processing [46][53]and pattern recognition [54][23], the use of cellular automata is very extensive. CA machines have a high degree parallelism. We can simulate complex systems with the help of CA parallelism [55]. These systems can be achieved at very low cost in comprising of others. Self replicating machines [56][57][58]are used to solve the computation problem, CA helps in that. Solution of NP Complete problem [59]can be achieved with self-replicating structures, and we call it satisfiability.

## 3.5 IMAGE PROCESSING

A cellular automaton is used to do computation problem. It's tough to create general-purpose system for real tasks with CA. We know images are made of pixels or cellular pixels. CA is used for the cellular processing. There are two platforms for image processing [46][53]. The first approach is Cellular logic array and the second approach is cellular neural network (cellular nonlinear network). The base of image processing is cellular logic and the algorithm base is cellular hardware. As we know cellular automata is based on local rules so it follows the same rule in image processing too. An image is decomposed by using CA local rules. CA is the way by which we can design fast image processing platform.

## 3.6 PSEUDORANDOM NUMBER GENERATORS

Cellular automata is also used to generate Pseudorandom Number [46][2]. We use controllable CA for this method. We can obtain the real random numbers by some physical phenomenon. Rule 30 [60] of cellular automata is use to generate the random numbers [60]. The sequence of numbers in PNRG does not always random, it can be determined by initial values of small cells. CCA (controllable CA) PNRG and PCA (programmable CA) PNRG are the two approaches for generating random numbers. PCA is non-uniform, different rules are allowed at different time stamps. Transition role of cell play a very major role in this whether it is uniform or non-uniform CA.

# 4. CELLULAR AUTOMATA AS A CLASSIFIER

The ***k*-nearest neighbor algorithm** (*k*-NN) is a method approach by which we can classify objects[61][62][63]. The basis of this method is closest training example. K-NN follows the instance based learning approach [64][65] or lazy learning in which the function acts locally and all calculation is delayed until classification. K-nearest neighbor is the easiest approach in machine learning algorithms. In machine learning algorithm [66][67] an instance is classified by its neighborhood instances with the instance being assigned to the behavior most common between its k neighborhood instances where k has integer value. If the value of k is one then the instance is assigned to its nearest neighbor class.K-NN method can be also used in regression in which the property value is assigned for the instance to be equal to the average value of the instance neighbors. This method can be also used to weight the contribution of its neighborhood instances so that the instances which are nearer gives more contribution to the average as compare to the distance neighbors. A weight of 1/d is given to the common weighting scheme where d is the distance between instance and its neighborhood instance. This approach is linear interpolation generalization.

Neighborhood instances are taken from pool of instances for which the exact classification is done. It can be taken as training set input for the algorithm and no external training action is required. K-NN algorithm is very good for the data's local structure.

Decision boundaries are computed by nearest neighbor rules implicitly and effectively. It can also be computed by itself in explicit manner and to do this efficiently because the computational complexity is a major method of the boundary complexity.

Vector with a class label are the training examples in multidimensional space. Algorithm's training part only contains the logic of storing the vector with a class label and in algorithm's classification part, we create a user defined constants k and k is a positive integer value and a vector without label is classified by allocating a label and this vector would be the most frequent vector between the user defined k samples.

Ibk classifier [66][68][69]is cellular automata based classifier that is based on pattern search and follows the K-NN based algorithm to classify the datasets.

# 5. DISCUSSIONS

We can create an effective classifier with cellular automata that use a pure local decision. If we use simple cells in a collective behavior then it achieves the classification performance. Each cell has its own local decision and work with the neighbor's information. We put the cells in the grid space in such a way that each attribute of the classified dataset will be in its own dimension. But we have to consider the grid size because it will increase with number of attributes of the dataset. So the k- nearest neighborhood method is one of the best approach for this type of classification.

# 6. EXPERIMENT RESULT

Here we take the experiment result of same data set known as IRIS taken from UCI repository .and applies different classifiers in Data Mining .In this dataset we have 5 attribute and 150 instances.

Analysis on that data set given below .

***A.*** Comparison and Analysis of different Classifiers in Data Mining (Iris Dataset)

| **Classifiers** | **Execu tion Time** | **Corre ctly Classi fied Insta nces** | **Incorr ectly Classifi ed Instan ces** | **TP Rate (Weig hted Avg.)** | **FP Rate (Weig hted Avg.)** |
|---|---|---|---|---|---|
| IBK (Lazy Classifier) | 0 secon ds | 95.33 33 % | 4.6667 % | 0.953 | 0.023 |
| J48(tree based classifier) | 0.02 secon ds | 96% | 4% | 0.96 | 0.02 |
| BayesNet(ba yes based classifier) | 0.02 secon ds | 92.66 67 % | 7.3333 % | 0.927 | 0.037 |
| Bagging(met a based classifier) | 0.03 secon ds | 94.66 67 % | 5.3333 % | 0.947 | 0.027 |
| SMO(functio ns based classifier) | 0.19 secon ds | 96% | 4% | 0.96 | 0.02 |
| VFI(misc based classifier) | 0 secon ds | 96 % | 4 % | 0.96 | 0.02 |
| DecisionTabl e(rules based classifier) | 0.02 secon ds | 92.66 67 % | 7.3333 % | 0.927 | 0.037 |

***Table 2***

All the above classifiers are implemented in Weka. IBK (Lazy Classifier) is a Cellular Automata based classifier. I studied different classifiers output and try to analyze their performance. Table 1 shows the result of different classifiers. Their execution time and accuracy. On the basis of above table , we can conclude that the most accurate classifiers are J48 and SMO because they correctly classified 96% instances. But SMO and J48 takes 0.19 seconds and 0.02 seconds to execute. So, on the basis of execution time J48 is better than SMO. On the other hand, IBK takes minimum time to execute that is 0 second and it is 95.3333% accurate. So, on the basis of execution time IBK(Lazy Classifier) shows comparatively better result and that is Cellular Autamata based Classifier.

***B.*** Comparison and Analysis of Cellular Automata (CA) Based Classifiers in Data mining (Iris Dataset)

| CA based Classifiers (Lazy Classifier) | Execution time | Correctly Classified Instances | Incorrectly Classified Instances | TP Rate (Weighted Avg.) | FP Rate (Weighted Avg.) |
|---|---|---|---|---|---|
| IB1 | 0 seconds | 95.3333 % | 4.6667 % | 0.953 | 0.023 |
| IBK | 0 seconds | 95.3333 % | 4.6667 % | 0.953 | 0.023 |
| KStar | 0 seconds | 94.6667 % | 5.3333 % | 0.947 | 0.027 |
| LWL | 0 seconds | 93.3333 % | 6.6667 % | 0.933 | 0.033 |

***Table 3***

Lazy Classifier is a CA based classifier. IB1, IBK, KStar and LWL are different types of Lazy Classifiers. I performed the comparative analysis work on different CA based classifiers in Weka. On the basis of Table 2 we conclude that all lazy classifiers execution time is almost same i.e zero seconds. Their accuracy is also almost same. But the least accurate one is the LWL which correctly classified only 93.3333% instances.

***C.*** Comparison and analysis of CA based Classifier on different datasets

| Dataset | IBK (Lazy Classifier) Execution Time |
|---|---|
| Abalone | 0 seconds |
| Adult | 0.01 seconds |
| Anneal | 0 seconds |
| Iris | 0 seconds |

***Table 4***

Table 3 shows the performance of CA based classifier i.e Lazy classifier (IBK) on different datasets. On the basis of Table 3 we can conclude that IBK takes almost equal time (0 seconds) on every dataset.

## 7. CONCLUSIONS

This paper elaborates the fusion of data mining with cellular automata. It tells the brief history of cellular automata and data mining systems. It also includes the different type of classification algorithm of data mining and their brief associative approaches.

It contains the cellular automata behavior with the instances of the class, how they behave, what are the characteristics of object and their respective neighbors. It also describes the uses of cellular automata in other areas such in biological, ecological and in parallel computing. We can say that with the pattern recognition approach and following the k-nearest neighborhood algorithm it is possible to achieve the classification through cellular automata that define a fusion of data mining with cellular automata. I think this paper will really help to cellular automata researcher in coming future.

## 8. REFERENCES

[1] Tom Fawcett (Center for the Study ), "Data mining with cellular automata," pp. 1-10, 2007.

[2] N. Ganguly, B. K. Sikdar, A. Deutsch, G. Canright, and P. P. Chaudhuri, "A Survey on Cellular Automata∗," *Engineering*, pp. 1-30.

[3] P. Gu and Y. Zheng, "A New Data Mining Model for Forest-Fire Cellular Automata," *2012 Fifth International Conference on Intelligent Computation Technology and Automation*, pp. 37-40, Jan. 2012.

[4] H. Nishio, "Changing the Neighborhood of Cellular Automata," no. September, pp. 10-13, 2007.

[5] S. Physics, R. October, K. E. Y. Words, and A. Study, "Two-Dimensional Cellular Automata," vol. 38, 1985.

[6] C. D. Brummitt, H. Delventhal, and M. Retzlaff, "Packard Snowflakes on the von Neumann Neighborhood," *Science*, vol. 3, pp. 57-79, 2008.

[7] P. M. Wocjan and C. Science, "Hamiltonian Cellular Automata in Joint work with Daniel Nagaj ( MIT )," *Electrical Engineering*.

[8] A. Seth, S. Bandyopadhyay, and U. Maulik, "Probabilistic Analysis of Cellular Automata Rules and its Application in Pseudo Random Pattern Generation," *International Journal*, no. November, 2008.

[9] B. Meshkboo and M. Kangavari, "Video Data Mining with Learning Cellular Automata," pp. 1-9.

[10] A. Sleit, A. Dalhoum, I. Al-dhamari, and A. Awwad, "Efficient Enhancement on Cellular Automata for Data Mining," *Technology*, pp. 616-620.

[11] L. Zhou and M. Yang, "A Classifier Build Around Cellular Automata for Distributed Data Mining," *2008 International Conference on Computer Science and Software Engineering*, pp. 312-315, 2008.

[12] T. N. Phyu, "Survey of Classification Techniques in Data Mining," *Computer*, vol. I, 2009.

***Table 3***

[13] R. M. Hamou, A. Lehireche, A. C. Lokbani, and M. Rahmani, "Text Clustering by 2D Cellular Automata Based on the N-Grams," *2010 First ACIS International Symposium on Cryptography, and Network Security, Data Mining and Knowledge Discovery, E-Commerce and Its Applications, and Embedded Systems*, pp. 271-277, Oct. 2010.

[14] M. P. and P. M. M. S. of G. U. of N. N. U. K. N. 2RD, "DECISION TREE BASED CLASSIFICATION OF REMOTELY SENSED DATA," no. November, pp. 5-8, 2001.

[15] P. Yin, A. Criminisi, J. Winn, and I. Essa, "Tree-based Classifiers for Bilayer Video Segmentation," *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1-8, Jun. 2007.

[16] J. N. Al-karaki and A. E. Kamal, "Routing Techniques in Wireless Sensor Networks : A," *Computer Engineering*, pp. 1-37.

[17] M. Modeling, "Different Routing Techniques in VANET."

[18] U. S. Office and P. Management, "THE CLASSIFIER ' S HANDBOOK Table of Contents ( Also See The Introduction to the Position Classification Standards .) THE CLASSIFIER ' S HANDBOOK Table of Contents ( Continued )," *Management*, vol. 41, no. August, pp. 1-45, 1991.

[19] K. P. Murphy, "Naive Bayes classifiers Generative classifiers," *Bernoulli*, pp. 1-8, 2006.

[20] J. V. Marcos, R. Hornero, D. Alvarez, F. del Campo, M. López, and C. Zamarrón, "Radial basis function classifiers to help in the diagnosis of the obstructive sleep apnoea syndrome from nocturnal oximetry.," *Medical & biological engineering & computing*, vol. 46, no. 4, pp. 323-32, May 2008.

[21] Y.-J. Oyang, S.-C. Hwang, Y.-Y. Ou, C.-Y. Chen, and Z.-W. Chen, "Data classification with radial basis function networks based on a novel kernel density estimation algorithm.," *IEEE transactions on neural networks / a publication of the IEEE Neural Networks Council*, vol. 16, no. 1, pp. 225-36, Jan. 2005.

[22] A. G. Bors, "Introduction of the Radial Basis Function ( RBF ) Networks," *Science*, pp. 1-7.

[23] C. J. C. Burges, "A Tutorial on Support Vector Machines for Pattern Recognition," *Data Mining and Knowledge Discovery*, vol. 43, pp. 1-43, 1997.

[24] S. Tong and D. Koller, "with Applications to Text Classification," *Journal of Machine Learning Research*, pp. 45-66, 2001.

[25] W. S. Noble and P. Street, "What is a support vector machine ?," *Nature Biotechnology*, vol. 24, no. 12, pp. 1565-1567, 2006.

[26] "A LAZY LEARNING CONTROL METHOD USING SUPPORT VECTOR REGRESSION Masayuki Kobayashi, Yasuo Konishi and Hiroyuki Ishigaki," *Information and Control*, vol. 3, no. 6, p. 2007, 2007.

[27] Z. Zheng and G. I. Webb, "Lazy Learning of Bayesian Rules," vol. 87, pp. 53-87, 2000.

[28] M.-ling Zhang and Z.-hua Zhou, "Ml-knn : A Lazy Learning Approach to Multi-Label Learning," *Technology*.

[29] G.-xun Yuan, C.-hua Ho, and C.-jen Lin, "Recent Advances of Large-scale Linear Classification," *Computer*, pp. 1-25.

[30] R.-en Fan, X.-rui Wang, and C.-jen Lin, "LIBLINEAR : A Library for Large Linear Classification," *Corpus*, vol. 9, pp. 1871-1874, 2008.

[31] M. Dredze, "Confidence-Weighted Linear Classification," *Update*, 2008.

[32] T. Weise, "Global Optimization Algorithms – Theory and Application –," 2009.

[33] M. V. Butz and M. Pelikan, "Studying XCS / BOA Learning in Boolean Functions : Structure Encoding and Random Boolean Functions Categories and Subject Descriptors," *Cognitive Psychology*.

[34] S. W. Wilson, "State of XCS Classifier System Research," *Environment*, no. 99, pp. 1-21, 1999.

[35] P. I. Armstrong, S. X. Day, J. P. McVay, and J. Rounds, "Holland's RIASEC model as an integrative framework for individual differences.," *Journal of Counseling Psychology*, vol. 55, no. 1, pp. 1-18, 2008.

[36] M. Abedini and M. Kirley, "Guided Rule Discovery in XCS for High-dimensional Classification Problems," *Engineering*.

[37] A. Orriols-puig, D. E. Goldberg, E. Bernad, and K. Sastry, "Modeling XCS in Class Imbalances : Population Size and Parameter Settings," no. January, 2007.

[38] A. Gandhe, S.-hsin Yu, and R. E. Smith, "XCS for Fusing Multi-Spectral Data in Automatic Target Recognition."

[39] Q. Jackson and D. Landgrebe, "No Title," *Office*, pp. 1-35, 2001.

[40] P. Sarkar, "A brief history of cellular automata," *ACM Computing Surveys*, vol. 32, no. 1, pp. 80-107, Mar. 2000.

[41] "Game of Life," *Physics*, 1970.

[42] G. P. Zhang, "Neural Networks for Classification : A Survey," vol. 30, no. 4, pp. 451-462, 2000.

[43] A. S. Introduction, "Neural Networks," *Neural Networks*, 1996.

[44] C. Gershenson, "Artificial Neural Networks for Beginners," *Networks*, pp. 1-8.

[45] M. Daven, A. Kheyfits, G. Zeleke, MSam, and Elkamu, "An introduction to Cellular Automata and their applications Pascal ' s Triangle," *Group*, pp. 1-23, 2001.

[46] N. Yu, M. Li, and X. Ruan, "Applications of cellular automata in complex system study," *Education*, vol. 1, no. 3, pp. 302-310, 2005.

[47] A. P. Carlos, "Quantum Cellular Automata : Theory and Applications by," 2007.

[48] J. R. Wolpaw, D. J. McFarland, and T. M. Vaughan, "Brain-computer interface research at the Wadsworth Center.," *IEEE transactions on rehabilitation engineering : a publication of the IEEE Engineering in Medicine and Biology Society*, vol. 8, no. 2, pp. 222-6, Jun. 2000.

[49] B. R. Friedman, "No Title," *Simulation*, 2009.

[50] K. D. Anthony, "Introduction to Causal Modeling , Bayesian Theory and Major Bayesian Modeling Tools for the Intelligence Analyst," *European Journal Of Operational Research*, pp. 1-31.

[51] P. S. Albin, "Rationality :"

[52] "Theory and Applications of Nonlinear Cellular Automata In VLSI Design Theory and Applications of

Nonlinear Cellular Automata In VLSI Design," *Engineering and Science*.

[53] I. Processing, "Image Processing," *Image Processing*, pp. 404-465.

[54] "What is Pattern Recognition ?," vol. 25, no. 1, 2003.

[55] D. Talia, "Cellular Automata + Parallel Computing = Computational Simulation," *Computer*.

[56] J. Suthakorn and G. S. Chirikjian, "A Semi-Autonomous Replicating Robotic System," *Mechanical Engineering*, pp. 776-781, 2003.

[57] A. Smith, P. Turney, and R. Ewaschuk, "Self-replicating machines in continuous space with virtual physics.," *Artificial life*, vol. 9, no. 1, pp. 21-40, Jan. 2003.

[58] I. Electricista, M. Automática, and U. Valle, "Implementation of a Self-Replicating Héctor Fabio Restrepo-García," vol. 2457, 2001.

[59] "NP-complete problems," *Search*.

[60] D. Gage, E. Laub, and B. Mcgarry, "Cellular automata: is rule 30 random?," pp. 1-10.

[61] G. Guo, H. Wang, D. Bell, Y. Bi, and K. Greer, "KNN Model-Based Approach in Classification," *Idea*.

[62] J. A. Gamez and P. Puerta, Jose MBermejo, "Improving KNN-based e-mail classification into folders generating class-balanced datasets .," pp. 529-536, 2008.

[63] A. Ault, X. Zhong, and E. J. Coyle, "K-Nearest-Neighbor Analysis of Received Signal Strength Distance Estimation Across Environments," *Environment*.

[64] A. D. Lattner, "Instance-Based Learning and Information Extraction for the Generation of Metadata," *Management*, pp. 472-479, 2003.

[65] D. W. Aha, D. Kibler, and M. K. Albert, "Instance-based learning algorithms," *Machine Learning*, vol. 6, no. 1, pp. 37-66, Jan. 1991.

[66] A. Y. Al-Omary and M. S. Jamil, "A new approach of clustering based machine-learning algorithm," *Knowledge-Based Systems*, vol. 19, no. 4, pp. 248-258, Aug. 2006.

[67] I. H. Witten and E. Frank, "Machine Learning Algorithms in Java Nuts and bolts : Machine," *Machine Learning*, 2000.

[68] G. Schmidberger, "Practical Data Mining Tutorial 2 : Nearest Neighbor Learning and Decision Trees," *Learning*, 2008.

[69] T. V. Inglesby and D. a Henderson, "Introduction.," *Biosecurity and bioterrorism : biodefense strategy, practice, and science*, vol. 10, no. 1, p. 5, Mar. 2012.