

IT Infrastructure Downtime Preemption using Hybrid Machine Learning and NLP

Chiranjiv Roy, Sourov Moitra, Rashika Malhotra, Subramaniyan Srinivasan, Mainak Das

Hewlett Packard, Technology Services, GSD CSC Bangalore, India
{c.roy, sourov.moitra, rashika.malhotra, ssn, mainak.das}@hpe.com

Abstract—IT Infrastructure Management and server downtime have been an area of exploration by researchers and industry experts, for over a decade. Despite the research on web server downtime, system failure and fault prediction, etc., there is a void in the field of IT Infrastructure Downtime Management. Downtime in an IT Infrastructure can cause enormous financial, reputational and relationship losses for customer and vendor. Our attempt is to address this gap by developing an innovative architecture which predicts IT Infrastructure failure. We have used a hybrid approach of human-machine interaction through Big Data, Machine Learning, NLP and IR. We sourced real-time machine, operating system, application logs and unstructured case notes into an algorithm for multi-dimensional symptoms mining, using iterative deepening depth-first search, traversal to create transactions for Sequential Pattern Mining of symptoms to events. It went through multiple statistical tests and review from technology experts, to create and update a dynamic Pattern Dictionary. This dictionary is used for training unsupervised and supervised classification models of machine learning, namely SVM and Random Forrest to score and predict new logs in a real time mode. The approach is also dynamic to use unsupervised clustering methods to give directions to the technicians on future or unknown pattern of errors or fault, to constantly update the Pattern Dictionary and improve classification for new IT products.

General Terms—Experimentation, Algorithms, Service Support, Technology, Research.

Index Terms—IT Infrastructure Management, Big Data, Data Mining, Natural Language Processing, Information Retrieval, Decision Tree, Early Warning System, Entropy, k Nearest Neighbor Classification Algorithm, Random Forrest Classifier, Support Vector Machine, Event Clustering, Naïve Bayes Classifier.

I. INTRODUCTION

AN UNPLANNED maintenance can be financially damaging. Marshall Institute, an asset management consultancy, reckons that a good rule of thumb is that breakdown or emergency repairs cost three times the cost of preventive, predictive or planned corrective maintenance, which still needs an end to end solution.

We, through this research introduce the concept of a holistic IT Infrastructure Downtime Management by opening a field of study to the future.

II. RELATED WORK

Research on existing literature and industry practices reveal that many companies have recently launched services on predictive maintenance of IT equipment. Almost all services

companies have some form of offering in this space. Specific book on predictive maintenance was written in early 2000 by R Keith Mobley. The literature suggests in general that we can use rule based thresholds on live monitored data of equipment and raise alerts based on rules. The literature also talks about multiple statistical and machine learning algorithms that can be used to classify new events. But the dependencies on human codified rules compel solutions to be more customized and require multiple human interventions to function. The approach presented in this paper recommends usage of multiple existing and new methods to ensure we have automated mechanisms, to build rules which then can be validated, if required. This makes the solution generic, usable across multiple types of equipment and across IT infrastructure across industries.

Regarding algorithms used most often existing solutions specify regression models to understand and forecast failures. There is also usage of binary and multiclass classification models to predict whether a failure will occur within a certain time frame. This is particularly apparent in Microsoft's Predictive Maintenance gallery model in Azure Machine learning platform. The implicit assumption in this type of usage of existing algorithms is that equipment sensor data is structured to tell us a sequence of events leading to failure. The approach in this paper suggests mechanism to work around this by identifying the pattern in sequence of events that lead to failures from raw machine data, where symptomatic events leading to final failure are not explicit in nature.

Another important area which was very apparent from research surveys but mostly missing in existing industry implementation of predictive maintenance is Gartner and IDC's 80-20 rule: 80% of all downtime is due to people and process issues and 20% due to technology issues. Over dependence on machine sensors constraints the model to look and predict a machine's performance. This leads to an oversight of people and process issues. Our work in this paper highlights that with generic pattern identification and subsequent classification based on validated patterns can help capture people and process issues as well.

Thus the paper tries to extend existing related work into generic predictive maintenance solution which is applicable to an environment with varied IT infrastructure. People and process issues are also predicted in this generic approach.

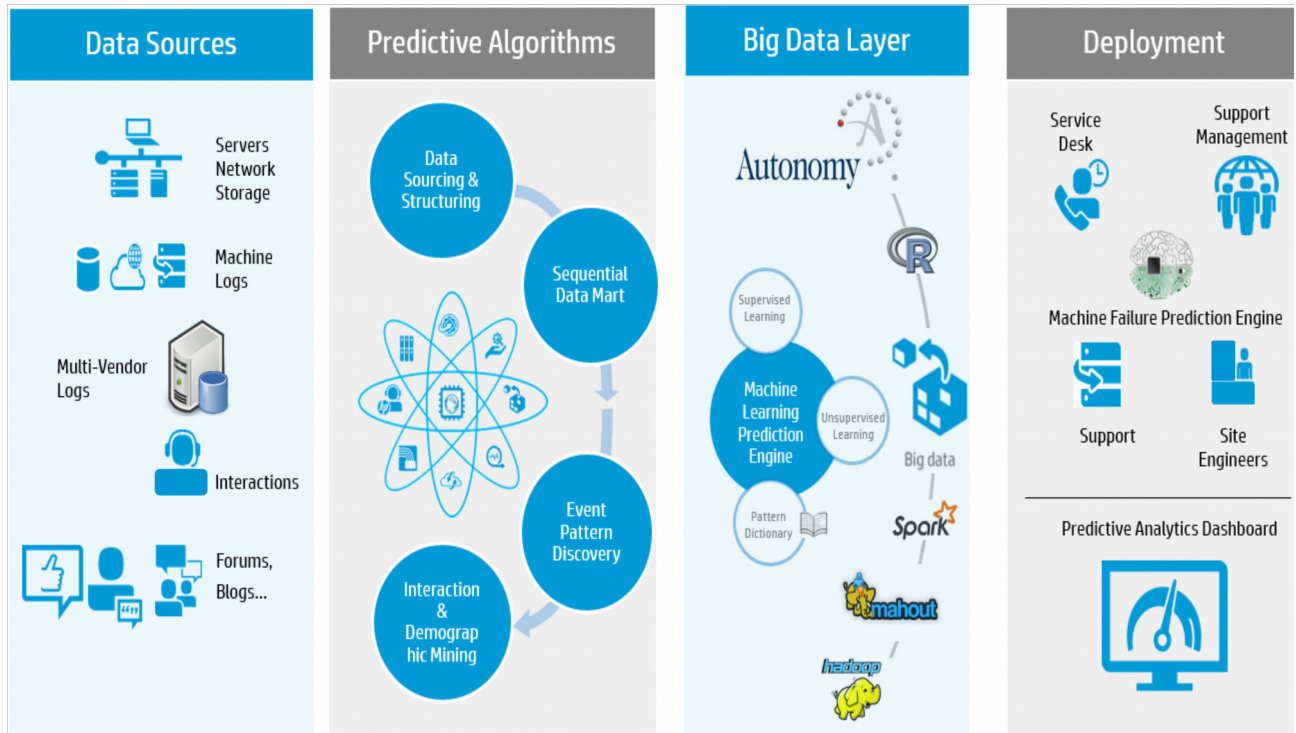


Fig 1. High Level Diagram

III. HIGH LEVEL DESIGN

Figure 1 gives an overview of our approach

IV. DATA SOURCING

Machine performance data is stored as Log, and is the main resource to understand machine behavior. This can be collected over network using programs written in different languages. An Enterprise IT environment uses different types of equipment based on the usage. The challenges are:

- Variety of the Product: Servers, Storage, Networking, power backups etc.
- Multi manufacturer: Same types of the product from different manufacturer placed in one environment.
- Types of machine Data: every machine has its own style of providing data. Some of them are plain text, some of them are encrypted or as a binary file with different types of data structure

The challenge is combining all of the data to one readable and usable format for the algorithm to understand and apply models for study and usage. Above all, the challenges of doing data mining and modeling on such large data are many. Statistically surmising, continuous and discrete data together has always been a challenge. In a huge dataset with multiple sources, with multiple pieces of information from independent sources, it is very difficult to infer something based on looking at just a few pieces of information; as different permutations of multiple pieces of information could lead to various conflicting events. The data is also dynamic, as logs are continuously fed. On top of that, the inflow of information is fast paced, huge and requires immediate attention to trigger corrective actions on time. So unless and until the snapshot of the data is

captured and analyzed with due consideration to timeline, it can lead to incoherent inferences.

The challenges don't just end there. The lack of centralized data base with multiple systems handling different type of data for different domains and with lack of proper governance across them, a lot of data problems like duplicity, missing data, wrong data, improper time tracking etc. also arise. The real challenge comes in tracing whether the data is correct or not. From analysis perspective, the "descriptions" for each line of log are captured as unstructured data; which if analyzed accurately using advanced text mining algorithms, along with Machine Learning Supervised methods, in an Hybrid Learning Method, can give great insights into the Downtime Management, and if missed can lead to erroneous conclusions.

V. DISCOVERING PATTERNS—THE SUPERVISED WAY

A. Developing Symptoms Database

Let us elucidate the need for self-learning Transactional Table and how we have conceptualized it. Each source of data becomes very dynamic post decrypting in terms of flow of events and defining relationships to be mined. We studied and found that an enterprise data of such a large scale does not have a standardized way to collapsing all the sources in a robust format for further study. The generic approaches as illustrated in [20][21][22][23][24][29] cannot be directly applied because of diverse systems resulting in complex behavior as seen in the logs. A sample as shown in table (1) illustrates the complexity in back and forth rapidly changing dimensions of the Symptoms to Events generation. The final result may also be skewed towards either a complete failure

TABLE I.
SYMPTOMS TABLE

Symptom 1 (S _i)	Symptom 2 (S ₂)	Symptom N (S _n)	Event
S ₁₁	S ₁₃	S ₅₂	E ₁
S ₂₁	S ₄₄	S ₅₇	E ₁
S ₁₇	S ₃₁	S ₆₆	E ₂

of the system or a symptom or an event, hence complexity multiplies itself.

B. Event Pattern Discovery

Pattern mining or discovery is a holistic concept and has been applied in Marketing Science [25][26] and other research fields [27] for decades now. Analyzing symptoms from billions of rows into meaningful patterns in rapidly changing environments like IT Infrastructure. We had to take a non-traditional way to approach this problem. We first created a framework of ‘‘Pattern Dictionary’’ which works in two ways. We developed a path based combined length and breadth search function as illustrated in papers in bits and pieces in studies [15][16][17][18][19]. The result of this exercise was to find the best possible pattern using multi-level search and precisely try to churn the duplicates at each hierarchical level of system, server or entire environment and find best known patterns.

The second way was to apply Constraint based Sequential Pattern Mining [22] [23] [24], to study and find the unknown patterns using pattern growth method for frequent pattern mining. The sequence of the data was determined by the transactions by finding events occurring at regular intervals and then finding difference between two similar events.

We then applied a visual Self Organizing Map on the rules to determine the rules generated with the in-house function and sequential pattern miner (example as shown in Figure 2). The resultant patterns are clubbed together and gets into the cleansing process.

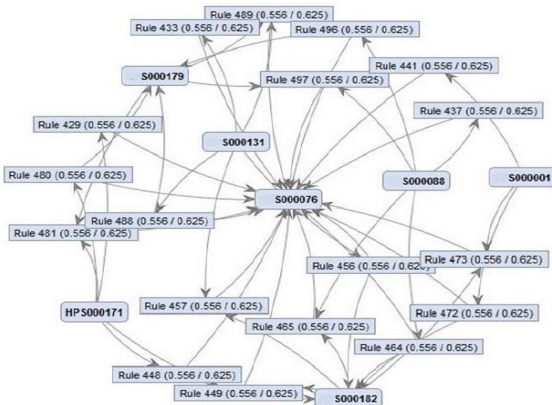


Fig 2. Rule Network using Self Organizing Map

A sequence $\alpha = (X_1 \dots X_l)$ is an ordered list of itemsets or symptoms in our case. An itemset X_i ($1 \leq i \leq l$) in a sequence is called a transaction, a term originated from analyzing customers’ shopping sequences in a transaction database.

The number of transactions in a sequence is called the ‘length of the sequence’. A sequence with length l is called an l -sequence. For an l -sequence α , we have $len(\alpha) = l$. Furthermore, the i -th itemset is denoted by $\alpha[i]$. An item can occur at most once in an itemset, but can occur multiple times in various itemsets in a sequence.

A sequence $\alpha = (X_1 \dots X_n)$ is called a subsequence of another sequence $\beta = (Y_1 \dots Y_m)$ ($n \leq m$), and β a super-sequence of α , denoted by $\alpha \preceq \beta$, if there exist integers $1 \leq i_1 < \dots < i_n \leq m$ such that $X_1 \subseteq Y_{i_1}, \dots, X_n \subseteq Y_{i_n}$.

A sequence database is a set of 2-tuples (sid, α) , where sid is a sequence-id and α a sequence. A tuple (sid, α) in a sequence database SDB is said to contain a sequence γ if γ is a subsequence of α . The number of tuples in a sequence database SDB containing sequence γ is called the support of γ , denoted by $sup(\gamma)$.

Given a positive integer min_sup as the support threshold, a sequence γ is a sequential pattern in sequence database ‘‘Patterns Database’’ if $sup(\gamma) \geq min_sup$. The sequential pattern mining problem is to find the complete set of sequential patterns with respect to a given sequence database SDB and a support threshold min_sup .

The first principle that we apply here is applying a weightage based on Pearson’s chi-square test, also known as the chi-square goodness-of-fit test or chi-square test for independence [32]. This gives an advantage over and above choosing the rules based on lift and support. The result from association rules P_1 passes on to the Human Machine Interface (HMI) for next level of check.

C. Human Machine Interface for Pattern Validation

Patterns from the mining activity as illustrated above is treated in the algorithm using a new and robust mechanism of throwing the first patterns P_1 to be matched with the generic list of error databases existent with all enterprise support organizations. The resultant is Pattern List 2 (P_2) which is then passed to the experts and technicians to validate the patterns with their experiences through a web portal in cloud to capture their validations using voting methodology of event occurrence. The weight of voting on specific pattern clears its path for the self-learning and ever evolving ‘‘Pattern Dictionary’’. This eliminates bias from statistical and industrial application perspective as shown in Table 2. A java based portal (J_p) is used to show the entire frame and the data is stored in the HP Vertica data center.

TABLE II.
SYMPTOMS TABLE

Pattern	Event	Lift	Vote	P _{Votes}
$S_{11} > S_{43} > S_{91}$	E ₁₁	0.69	Yes	39%
$S_{43} > S_{56} > S_{45} > S_{67}$	E ₂₃	0.35	No	77%

VI. PREDICTING THE NEXT EVENT—SUPERVISED MACHINE LEARNING APPROACH

Now that the “Pattern Dictionary” is ready with the patterns from the exercise done in Step 5, it is then needed to predict the next event occurrence. Decision Trees, [13] classically try to address this problem at a smaller scale and dimension (including limited classes). But at a big data level, we thought of applying Machine Learning in a different way on results from Association rule applied in some studies[1][2][3][4][5][6][7][8][9]. We have tried to study some machine learning models which is one the way of being implemented using Python and Spark.

A. Application of Machine Learning models with supervision

Every generation of machines may have a different behavior and may generate varied class of events. Using the logic of determining the best fit model from historical data may collapse when applied to real time. Hence we approached this problem by providing the flexibility of choosing the best fit model for a particular one real time, using a Java based portal (J_p).

We applied few machine learning models like kNN, Naïve Bayes, Support Vector Machine and Random Forest which can be scaled to Big Data framework at run time using Spark or Mahout (development in progress).

The above approach is quite intuitive in nature to capture complexities of the framework and give an advantage to the user or technician to apply the best fit model on real time. The first cut of results are shown in Table 3 above which tells us that the Sigmoid Kernel function used for SVM tends to give a better result but not far behind is the Gradient boosting model with 52 % classification accuracy. As there is

no existing model to test with, this testing model would be taken as reference for further evaluation.

B. Determining unknown events through Text Clustering and NLP

In the process of applying supervised mechanisms to predict next events we found that a significant percentage of events could not be classified or be predicted due to historical novelty, new machines developed or implemented, software updated, patch released etc. These may create a huge pile of symptoms which could not even be mined or mapped by human intelligence.

We propose a dynamic flexibility to the algorithm which takes the key aspects of Text Mining, Natural Language processing and clustering documents.

The first challenge was to put the millions of unstructured and structured data generated through server monitoring and collected through case logs, into a centralized data repository, on top of which an analytical system can be built. It was essentially a Big Data challenge, for which a HP Vertica Platform was used, to address it. Once the data was available, the next challenge was to apply machine learning on this Big Data. R MapReduce code came handy here to generate Document Term Matrix and Mahout was used for Clustering and Classification [31][32].

A clear roadmap was laid and followed to develop a Hybrid Pattern Mining and Unknown Issue Identification System. The incidents (symptoms) were divided into two sets: one where the affected items were mentioned and the other where the affected items were not mentioned. The incidents where the affected items were mentioned were further divided into the training and validation data sets. The training set was used to derive the affected items for incidents where it was not mentioned using machine learning (Pattern Min-

TABLE III.
SAMPLE RESULTS FROM MACHINE LEARNING MODELS

Selected Model	Predecessor Node	Model Node	Model Description	Target Variable	Misclassification Rate	Squared Error	Actual: Misclassification Rate
Y_{Event_A}	SVM	SVM	SVM-d	Final_Event	0.13582	0.13509	0.13582
	SVM3	SVM3	SVM-FQP	Final_Event	0.13582	0.13509	0.13582
	HPNNA2	HPNNA2	Select NN	Final_Event	0.13583	0.11834	0.13583
	HPNNA	HPNNA	HP Neural	Final_Event	0.13583	0.11988	0.13583
	LARS	LARS	LARS-d	Final_Event	0.13666	0.11319	0.13666
	SVM2	SVM2	SVM-LSVM	Final_Event	0.13674	0.14751	0.13674
	Reg	Reg	Regression-d	Final_Event	0.13682	0.11329	0.13682
	SVM4	SVM4	SVM-Sigmoid Kernel	Final_Event	0.54437	0.26851	0.54437
	Boost2	Boost2	Gradient Boosting - hash1	Final_Event	0.52273	0.52273	0.52273
	Boost	Boost	Gradient Boosting -d	Final_Event	0.13582	0.11712	0.13582

ing). The validation data set was used for in-sample validation. Once the affected items were known for all the incidents, the next step was to group those millions of independent incidents into logically related units which can be used to design an Incident Routing Mechanism. This mechanism ensures that a randomly reported incident is correctly allocated to its designate Level II tech support team on time, for accurate and timely resolution of the issue.

The Pattern Mining was done independently on a sample of top 5 clients by incidents/issues logged by call volume collected the free form textual inputs given by multi-level support representatives and server logs generated for those accounts connecting logs with R-Vertica Connect. First data cleansing was done including removal of punctuations, white spaces, numbers, stop-words, etc. The stop-word removal was a two-step process: first a standard corpus based and second specific to each account to take out many English word which are required for NLP but not needed in this case. The account based list was created. Next POS (parts-of-speech) tagging was done and only nouns and verbs with alphanumeric keys were kept. The alphanumeric keys was identified using N-gram taxonomy created and verified by experts. Finally, after all the cleansing, the DTM (Document Term Matrix) was created to link the term to the affected items. The DTM was shared with the engineers or experts to further refine it. The industry best practice of 10-fold cross validation technique was then used to find the relation between the unstructured incidents listed in the form free text and the affected items. The final DTM list was divided in training (90%) and validation (10%) sets, randomly 10 times. The kNN (k-nearest-neighbor) classification algorithm was run every time on the training set and the validation set was used to find the misclassification rate (error). The desired value of k (from 1 to 10) was selected based on the classification which gave the least misclassification rate (error). After the in-sample validation was done and we got both the right value of k and the relation between the unstructured incidents and the affected items; the findings were used to derive the affected items of those incidents where the affected items were not listed using kNN classification again. The output was a complete list of all the incidents with affected items.

The event determining mechanism was more challenging as there was no predefined/business logic behind grouping of similar incidents together and it was purely based on unsupervised machine learning algorithm. The k-means clustering algorithm of Mahout was chosen for the purpose. The biggest challenge here was to find the correct value of k. Silhouette approach was used to find the right value of k. In this method the items within each cluster are evaluated for their average dissimilarity with all the other items in the cluster. The measure of dissimilarity was selected as Euclidian Distance, say $a(i)$. The item is similarly evaluated for its average dissimilarity with any other cluster to which it does not belong, say $b(i)$. The cluster with the lowest average dissimilarity is said to be the 'neighboring cluster'. If $a(i) < b(i)$, then it can be concluded that the item belongs to the right cluster, else the clusters are revised. Thus the Silhouette approach is used to determine the value of k based on how tightly grouped all the items in a cluster are.

The choice of Silhouette method posed another challenge as it requires the data to be in the form of a non-negative square matrix to be run on Mahout. SVD (Singular Value Decomposition) technique was used to address this issue [28]. The DTM for the entire data set was created the same way as specified in the pattern mining section above, but for the use of Mahout instead of R. The DTM was transformed using SVD which not only resulted in a square matrix but also brought in dimensionality reduction; reducing the space complexity of the Silhouette algorithm which becomes highly space complex for Big Data. Once the logical grouping of incidents based on terms identified from the unstructured data in the form of comments for each incident was completed, the routing mechanism to decide which should be routed to which support team was very easy to develop.

These patterns are displayed online using the portal to map closely related events and results are saved in the Pattern dictionary.

C. Bridging the Supervised and Unsupervised approach

The "Pattern Dictionary" acts as the stakeholder for collecting information from both the supervised and unsupervised methods. These patterns may not be conclusive but are evolutionary in nature which develops as time grows and studies each system working as a perfect interface between Human and Machine.

The approach collects all the necessary pattern into the pattern dictionary and then applies multiple models to be tested as we do not know which model will fit better in what kind of situation.

Historical knowledge and Topics gained from the NLP approach makes this concept unique and a perfect bridge between Supervised and Unsupervised Methods.

VII. CONCLUSION

This research with Predictive Analytics in the center, is an attempt to preempt failures in a customer's IT environment using an algorithm with an approach starting with Data sourcing & structuring, transforming it for unique pattern discovery using the theory of Ripple effect from Symptoms, to Events for any IT infrastructure. This also uses external information like weather forecasting, blogs, customer surveys, social media to strengthen the hybrid machine learning algorithm. This is a scalable Big Data Solution, which predicts real time faults and downtime.

The benefit of this is that solutions are many including - prevention of unplanned downtime, maximizing the value of IT investments by reducing the cost, improving operational Efficiency, enabling Agility and Innovation, improving IT stability - less complexity and risk.

With this novel approach of using machine learning in a real world problem with multiple complexities, it tends to open a new field of study with integrated machines, servers, equipment and its impact on an overall IT environment.

VIII. FUTURE WORK

Due to the rapidly changing dynamic of the IT industry and the challenges faced with IT Infrastructure Management

as well as IT Infrastructure Maintenance, which is a billion of dollar industry in itself, it is the need of the hour to conduct research and support it. A novel amalgamative concept is what we have tried and briefly presented here.

We are still in the phase of learning as there is no research done so far on this field, hence the next phase is to try and first complete the solution in a big data scale after multiple testing cycles to make it robust enough for developing it as a solution for the first version.

The area we haven't explored much is the relationship mining of customer interactions and machine. Also optimizing it using Neural Network or Genetic Optimization techniques.

REFERENCES

- [1] Aggarwal, Charu C., Yu, Philip C. 2001. Outlier Detection for High Dimensional Data, ACM SIGMOD
- [2] Ghose, Udayan., Rai, C.S., Singh, Yogesh. 2010. On Multiplicative Entropy and Information gain in Large Data Sets, International Journal of Engineering Science and Technology, 187-193.
- [3] Han, Jiawei., Kamber, Micheline., Pei, Jian. 2011. Data mining: Concepts and Techniques, 561-562, Morgan Kaufmann.
- [4] Hodge, Victoria J., Austin, Jim. 2004. A Survey of Outlier Detection Methodologies, In: Artificial Intelligence Review, 85-126, Kluwer Academic Publishers, Netherlands.
- [5] Knorr, Edwin M., Ng Raymond T. 1998. Algorithms for Mining Distance-Based Outliers in Large Datasets, VLDB Conference.
- [6] Minka, Thomas P. 2003. A comparison of numerical optimizers for logistic regression.
- [7] Pawling, Alec., Chawla, Nitesh V., Chaudhary, Amitabh. 2005. Computing Information Gain in Data Streams, Temporal Data Mining Workshop.
- [8] Pliner, Vadim. 2004. A SAS® Macro for Naïve Bayes Classification.
- [9] Pokrajac, Dragoljub., Lazarevic, Aleksandar., Latecki, Longin Jan. 2007. Incremental Local Outlier Detection for Data Streams, IEE Symposium on Computational Intelligence and Data Mining (CIDM).
- [10] Rokach, Lior, Maimon, Oded. 2010. Decision Trees. In: Data Mining and Knowledge Discovery Handbook, 165-192, Springer.
- [11] Sahami, Mehran. 1996. Learning Limited Dependence Bayesian Classifiers.
- [12] Tan, Pang-Ning., Stienbach, Michael., Kumar, Vipin. 2007. Introduction to Data Mining, 139-20, Pearson.
- [13] Agrawal, R., Amielinski, T., and Swami, A. (1993). Mining association rule between sets of items in large databases. In Proceeding of the 1993 ACM SIGMOD International Conference on Management of Data, pp. 207-216, Washington, DC, May 26-28.
- [14] Agrawal, R. and Srikant, R. (1994). Fast algorithms for mining association rule. Proceedings of the 20th International Conference on Very Large Data Bases. pp. 487 – 499.
- [15] Antonie, M., Zañane, O. R., Coman, A. (2003). Associative Classifiers for Medical Images. Lecture Notes in Artificial Intelligence 2797, Mining Multimedia and Complex Data, pp 68-83, Springer-Verlag.
- [16] Blackmore, K. and Bossomaier, T. J. (2003). Comparison of See5 and J48.PART Algorithms for Missing Persons Profiling. Technical report. Charles Sturt University, Australia.
- [17] Brin, S., Motwani, R., Ullman, J., Tsur, S. (1997). Dynamic Itemset Counting and Implication Rules for Market Basket Data. Proceedings of the 1997 ACM SIGMOD International Conference on Management of Data.
- [18] Cendrowska, J. (1987). MODEL: An algorithm for inducing modular rules. International Journal of Man-Machine Studies. Vol.27, No.4, pp.349-370.
- [19] Cohen, W. W. (1995). Fast effective rule induction. In the Proceeding of the 12 th International Conference on Machine Learning, Morgan Kaufmann, San Francisco, pp. 115-123.
- [20] Cohen, W. W. (1993). Efficient pruning methods for separate-and-conquer rule learning systems. In the proceeding of the 13th International Joint Conference on AI, Chambry, France.
- [21] Cowling, P. and Chakhlevitch, K. (2003). Hyperheuristics for Managing a Large Collection of Low Level Heuristics to Schedule Personnel. Proceeding of 2003 IEEE conference on Evolutionary Computation, Canberra, Australia, 8-12 Dec 2003.
- [22] Dong, G., Li, J. (1999). Efficient mining of frequent patterns: Discovering trends and differences. In Proceeding of SIGKDD 1999, San Diego, California.
- [23] Chris Buckley and Alan F. Lewit, Optimizations of inverted vector searches, SIGIR '85, Pages 97-110, 1985..
- [24] Fayyad, U. M.; Piatetsky-Shapiro, G.; Smyth, P. (1996). Advances in knowledge discovery and data mining, MIT Press.
- [25] Zaki, M. J., Parthasarathy, S., Ogihara, M., and Li, W. (1997). New algorithms for fast discovery of association rules. 3rd KDD Conference, pp. 283-286, August 1997.
- [26] Charu C. Aggarwal, Stephen C. Gates and Philip S. Yu, On the merits of building categorization systems by supervised clustering, Proceedings of the fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Pages 352 – 356, 1999.
- [27] Paul Bradley and Usama Fayyad, Refining Initial Points for K-Means Clustering, Proceedings of the Fifteenth International Conference on Machine Learning ICML98, Pages 91-99. Morgan Kaufmann, San Francisco, 1998
- [28] Alvarez, Sergio A. Technical Report BC-CS-2003-01, July 2003. Chi-squared computation for association rules: preliminary results