Australian Journal of Basic and Applied Sciences, 10(12) July 2016, Pages: 117-125

AUSTRALIAN JOURNAL OF BASIC AND APPLIED SCIENCES

ISSN:1991-8178 EISSN: 2309-8414
Journal home page: www.ajbasweb.com

A Dynamic Neural Feature Selection Model using Cloud Databases

[1]R. Sridevi and [2]Dr. P. Dinadayalan

[1]*Research Scholar Bharathiar University Coimbatore, India.*
[2]*Department of Computer Science* Mahatma Gandhi Government Arts College Mahe, *India.*

Address For Correspondence:
R. Sridevi, Research Scholar Bharathiar University Coimbatore, India
E-mail: srideviphd@hotmail.com

ARTICLE INFO

Article history:
Received 26 April 2016
Accepted 21 July 2016
Published 30 July 2016

Keywords:
Recurrent Neural Network, Kohonen Network, Cloud Database, Associative Memory

ABSTRACT

"Dynamic Neural Feature Selection Model using Cloud Databases" is based on Recurrent Neural Network approach for Feature Selection which removes irrelevant, redundant, noisy data and retains relevant, correlated features in Cloud environment with Feature Interaction. The proposed model consists of Dynamic Self-organizing Feature Map and Feature Associative Memory. Dynamic Self-Organizing Feature Map implements the concept of Kohonen Self-Organizing map which uses Recurrent Neural Network with Feedback connections. The Dynamic Self-Organizing Feature Map clusters relevant and irrelevant features in the forward phase and removes noisy data in the reverse phase. The Feature Associative Memory intends to achieve Feature Interaction and removes redundant features. The experimental result shows that "Dynamic Neural Feature Selection Model using Cloud Databases" is better than the Conventional models.

INTRODUCTION

Cloud Computing is the delivery of computing as a service rather than a product, whereby shared resources, software and information are provided to users over the Internet. The service models of Cloud Computing are: Cloud Software as a Service (SaaS), Cloud Platform as a Service (PaaS) and Cloud Infrastructure as a Service (IaaS). In the Software as a Service, we focus only on Database as a Service. Database as a Service (DBaaS) is a cloud-based approach to the storage and management of structured data. DBaaS provides a flexible, scalable, on-demand platform that is oriented toward self-service and easy management.

Massive growth in digital data, changing data storage requirements, better broadband facilities and Cloud Computing led to the emergence of Cloud Databases. A Cloud Database is a database that typically runs on a Cloud Computing platform. In one of the deployment model of Cloud Database, users can purchase access to a database service, maintained by a Cloud Database provider. Among the SQL-based and NoSQL data model based databases available on the cloud, we use NoSQL database to store the dataset. A NoSQL database, also called Not Only SQL, is an approach to data management and database design that is useful for very large sets of distributed data. NoSQL is especially useful to analyze massive amounts of unstructured data or data that's stored remotely on multiple virtual servers in the cloud.

Feature Selection is the technique for identifying the most important features for learning is the base to machine learning. Learning performance can be highly increased by reducing the dimensionality of the data using Feature Selection. Real-world data, which is the input of the Feature Selection methods, are majorly affected by the presence of noise. There are two noise types namely Class (label noise) and attribute noise. The efficiency of the Feature Selection may be decreased due to the presence of noisy data in the data set. Some of the existing Feature Selection methods like Relief (Kira and Rendell, 1992), LVF (Las Vegas Filter) (Liu and Setiono, 1996) and ABB (Automatic Branch and Bound) (Liu *et al.*, 1998) handles noisy data using different

To Cite This Article: R. Sridevi and Dr. P. Dinadayalan., A Dynamic Neural Feature Selection Model using Cloud Databases. ***Aust. J. Basic & Appl. Sci.,*** *10(12): 117-125, 2016*

Australian Journal of Basic and Applied Sciences, 10(12) July 2016, Pages: 117-125

measures. Many of the Cloud based NoSQL data models did not support Feature Selection methods which handle redundant, irrelevant features and the noisy data present in the features.

Most of the conventional Feature Selection methods fail to remove redundant, irrelevant features; noisy data exists in the feature and interact with features. Some Feature Selection algorithms identify redundant and irrelevant features while considering Feature Interaction but fail to handle noisy data. Though some of the conventional Feature Selection methods handle the noisy data efficiently there is no chance of recursive paths in these methods. The level of noisy data present in the data can be reduced through the recursive paths provided by recurrent neural network. A neural network is a mathematical model or computational model that tries to simulate the structure and functional aspects of biological neural networks. The neural network demonstrates the behaviour of the feedforward network and recurrent network. Recurrent neural network have feedback path from their outputs to the inputs, the response of such network is recursive.

Thus, we propose a model “Dynamic Neural Feature Selection Model using Cloud Databases” which retrieves dataset from the Cloud Database namely Amazon’s AWS (Amazon Web Services). This new model identifies relevant, irrelevant features, removes noisy data present in the features and also interact among the features using recurrent neural network.

The rest of the paper is organized as follows. Section 2 describes the related works and their limitations. Section 3 proposes a new Feature Selection Model and its Components. Section 4 shows the experimental results in the form of tables and graphs. Finally, Section 5 summarizes proposed work and suggests future research directions.

Related Works:

A. Techniques and approaches in Feature Selection Methods:

Since 1970 Feature Selection has been an interesting research topic, more and more research work has been published. Wrapper approach and Filter approach are two commonly used approaches followed in Feature Selection. In Wrapper approach features are selected as part of the mining algorithm. In Filter approach features are selected before a mining algorithm, using heuristics based on general characteristics of the data, rather than a learning algorithm to evaluate the merit of feature subsets.

Of the existing algorithms, Branch and Bound (B&B) (Narendra and Fukunaga, 1977) is an optimal search algorithm in which exhaustive search can be avoided by using a monotonic measure. In this algorithm given threshold β (specified by user), the search stops at each node the evaluation of which is lower than β, so that efferent branches are pruned and can guarantee an optimal subset.

Traditionally, the research work on feature subset selection has focused on searching for relevant features. FOCUS (Almuallim and Dietterich, 1991) algorithm starts evaluating each singleton feature set, and then each set of two features and so forth. This method halts whenever a sufficiently consistent solution is found. This algorithm can detect redundant features, but not when data is noisy. An improvement on FOCUS algorithm introduces the concept of conflict between positive and negative examples to prune the search called FOCUS/2. This algorithm implements the MIN-FEATURE bias which is substantially faster than FOCUS and also handles noise-free training data only.

Feature weighting/ ranking algorithms (Kira and Rendell, 1992) weigh features based on different measures and rank them according to their relevance to the target concept. The Relief algorithm (Kira and Rendell, 1992) and its improved version Relief-F method weighs each feature based on a Euclidian distance measure and handles noisy and incomplete datasets, but still fails to handle redundant features. Floating Search methods in Feature Selection (Pudil *et al*., 1994) proposed two sequential search methods namely Sequential Backward Floating Selection (SBFS) and Sequential Forward Floating Selection (SFFS) in which number of features are included or eliminated at each step.

Mutual information is used as the evaluation function to identify the relevant features in Mutual Information Feature Selection (MIFS) and in the calculation of mutual information a weight coefficient is used to handle the redundant features. Conditional Mutual Information Maximization (CMIM) (Fleuret, 2004) is a very fast binary Feature Selection technique based on conditional mutual information which iteratively selects features on the condition of the features already selected. The Feature Selection method namely Consistency efficiently removes redundant features using certain strategy which searches for the minimal subset that separates different target concepts as consistently as the full set.

In Automatic Branch & Bound (ABB) (Liu *et al*., 1998) a new measure is employed which is monotonic and fast to compute. It is a Branch and Bound algorithm with its bound set to the inconsistency rate δ of data set with the full set of features and it removes irrelevant, redundant and finds correlated features even with the presence of noise. A hybrid algorithm (Dash and Liu, 1998) Quick Branch & Bound (QBB) composed of LVF (Liu and Setiono, 1996) and ABB (Liu *et al*., 1998) is more efficient than the Feature Selection methods (Almuallim and Dietterich, 1991; Liu and Setiono, 1996; Liu *et al*., 1998) in terms of average execution time and selected solution.

Koller and Sahami (1996) given an efficient algorithm for Feature Selection which computes an approximation to the optimal Feature Selection criterion and aims to remove both irrelevant and redundant features and states that two highly correlated feature will be both highly weighted. Based on the assumption that a good feature subset is one that includes features highly correlated with the target concept, yet uncorrelated with each other, a method namely Correlation based Feature Selection (CFS) was developed. This method evaluates the feature subset on the whole rather than individual ones. Fast Correlation Based Filter (FCBF) is correlation-based method for Feature Selection through relevance and redundancy analysis. This method addresses explicitly the correlation between features. Feature Selection Based on Association Rules (FSBAR) is used to find the features that are closely correlative with the class attribute by association rules mining method.

ElAlami (2009) proposed Genetic Algorithm to select feature subset from trained neural network namely Filter Model based on GA. This genetic algorithm is very effective in reducing dimensionality, removing irrelevant features but incapable of removing redundant Features. Multi-way interaction among features is achieved in Simulated Annealing Genetic Algorithm (SAGA) is designed for selecting optimal feature subsets efficiently with the ability to avoid being trapped in a local minimum of simulated annealing with the very high rate of convergence of the crossover operator of genetic algorithms. Clustering-based Feature Selection algorithm namely (CBFS) divides the features into several clusters, and selects the most representative feature from each cluster to form a subset of features.

Interaction among the features were first introduced by Guyon and Elisseeff (2003) describes filters that select variables by ranking them with correlation coefficients. Jakulin and Bratko (2004) introduced an operational definition of a generalized n-way interaction and observe both negative and positive interaction by using KIRKWOOD superposition approximation model for constructing part-to-whole approximation. INTERACT (Zhao and Liu, 2009) is a filter algorithm of backward elimination with c-contribution for finding interacting features. FOIL Rule based Feature Selection (FRFS) first merges the features appeared in the antecedents of all FOIL rules, uses a coverRatio metric in a candidate feature subset which excludes redundant features and reserves interactive ones. Interaction Weight based Feature Selection algorithm IWFS redress the traditional relevance measure between a feature and the class through the manipulation of interaction weight factor and ranks the candidate features with the adjusted relevance measure. This algorithm can deal with irrelevant, redundant and interactive features.

B. Neural network approach in Feature Selection:

Dea *et al*. (2001) use a technique from the field of information theory to select a set of important attribute and a neural network is trained which is used to classify tuples. Feed forward neural network is used to identify the salient features in a neural network based approach proposed by Verikas and Bacauskiene (2002). In the technique proposed by them, neural network is trained by minimizing the cross-entropy error function and this technique eliminates the least salient feature according to the ranking function.

A new wrapper Feature Selection approach using neural network (Monirul Kabir *et al*., 2010) employs three layered feed-forward neural network as a learning model. Recurrent Neural network is used in the Foreign Currency Exchange Rate Prediction (Rehman *et al*., 2014) along with CGP (Cartesian Genetic Programming). The prediction accuracy is highly enhanced in this recurrent neural network approach by increasing the number of feedback paths.

Hodge *et al*. (2012) introduce a unified framework for attribute selection and prediction. They introduced the CFS (Correlation-based Feature Selection) attribute selector into the AURA (hetero-Associative Memory neural network) k-NN (k-nearest neighbour) framework. Rajeswari *et al*. (2012) proposed a method for selecting important features using Artificial Neural network. In this method, MLP (Multi Layer Perceptron) is used for Feature Selection and checks accuracy using the Feed Forward Neural Network classifier after removal of every feature.

C. Categorization of Feature Selection Algorithms:

Over the past four decades, many Algorithms were designed for feature subset selection which can handle noisy data, removes redundant features, identifies and removes irrelevant features. So, the existing feature subset selection algorithms can be categorized as shown in Table I.

Table I: Categorization Of Feature Selection Algorithms

Category	FSA
Noise Tolerant	Relief, Relief-F,ABB, LVF
Eliminate Redundant Features	FOCUS, FCBF, Consistency, mRMR
Eliminate Irrelevant Features	Filter Model based on GA
Eliminate Redundant & Irrelevant Features	MIFS, FSBAR, ABB, CBFS,FRFS,IWFS
Feature Interaction	SAGA, INTERACT, FSBAR,FRFS,IWFS

Australian Journal of Basic and Applied Sciences, 10(12) July 2016, Pages: 117-125

Architecture For Dynamic neural Feature Selection Model:

From the literature survey, the Feature Selection methods (Almuallim and Dietterich, 1991; Peng *et al*., 2005) like FOCUS, FCBF, Consistency and mRMR removes redundant features but fails to handle noisy data. The Conventional Feature Selection methods (Kira and Rendell, 1992; Liu and Setiono, 1996) like Relief and LVF used different approaches to handled noisy data but cannot be able to handle redundant features. Filter Model based on GA algorithm (ElAlami, 2009) removes Irrelevant features but incapable of removing redundant features. Feature Selection algorithms (Liu *et al*., 1998) like MIFS, ABB and CBFS eliminate redundant and irrelevant features but fail to interact among the features. Feature Interaction is supported by Feature Selection methods (Zhao and Liu, 2009) like SAGA, INTERACT, FSBAR, FRFS, IWFS but these methods still fails to handle noisy data. The existing neural network approaches (Dea *et al*., 2001; Hodge *et al*., 2012; Monirul Kabir *et al*., 2010; Verikas and Bacauskiene, 2002) used in Feature Selection methods did not solve redundant, Feature Interaction and irrelevant features. Thus the traditional Feature Selection algorithms (Koller and Sahami, 1996; Liu *et al*., 1998; Pudil *et al*., 1994) fail to handle Feature Interaction and correlated features and also fail to remove irrelevant and redundant features with noisy data in Cloud Database environment.

To overcome the limitations of the conventional Feature Selection methods, we propose a new Feature Selection model, "Dynamic Neural Feature Selection Model using Cloud Databases", which removes irrelevant, redundant features, cloud noisy data and identifies correlated features with Feature Interaction in the cloud environment. This model incorporates the concepts of Artificial Neural Networks, Feature Selection methods and Cloud Databases. The Artificial Neural Network uses recurrent (dynamic) neural network, supervised and unsupervised training algorithms. NoSQL cloud datasets is implemented in the proposed model.

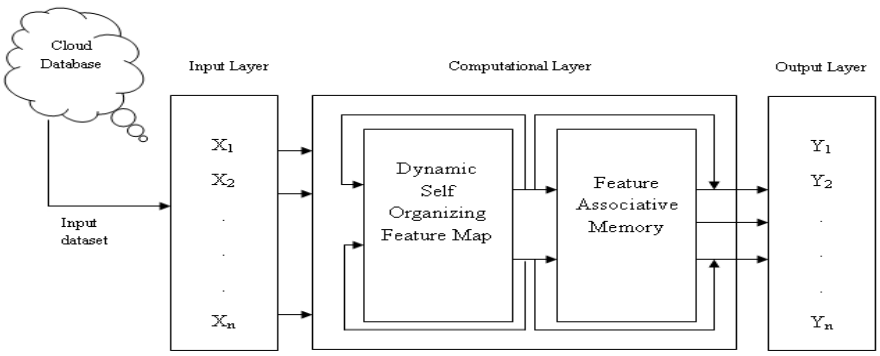

Fig. 1: Architecture of Dynamic Neural Feature Selection Model

Fig. 1 demonstrates the architecture of Dynamic Neural Feature Selection Model which has Input layer, Computational Layer and Output Layer. Input vectors are derived from the Cloud Database which is used as features. Let Schema T= $\{A_1,A_2\ldots A_n\}$ consists of set of attributes or features $\{A_1, A_2\ldots A_n\}$ from Cloud Databases. The types of features are Continuous, Integer and Nominal. The continuous type values are both real and Integer. The Nominal type values are taken from numeric as well as alphabets. The '-'and '?' represents 'not applicable' and 'missing values' respectively. Let $X = \{X_1, X_2\ldots X_n\}$ be the instance of schema T which are used as input vectors. The Output layer $Y = \{Y1, Y_2\ldots Y_n\}$ are the set of target vectors. The Computational layer consists of Dynamic Self-Organizing Feature Map and Feature Associative Memory. The Dynamic Self-Organizing Feature Map provides the solution for irrelevant, incorrect or incomplete and correlated features. The Feature Associative Memory offers the solution for Feature Interaction and redundant features.

D. Dynamic Self-Organizing Feature Map:

Dynamic Self-Organizing Feature Map component implements the concept of Kohonen Self-Organizing map in Recurrent Neural Network with feed-back connections. The structure of Dynamic Self-Organizing Feature Map is dynamic which is capturing the dynamics of successions by cycles in the network. It is based on unsupervised and competitive learning. The Dynamic Self-Organizing Feature Map provides solution for irrelevant, correlated features and feature with noisy data in cloud environment. The irrelevant, correlated features and features with noisy data are calculated using the Euclidean distance formula.

$$d(t+1) = d(t)\ \ w(t) + | X - Y| \qquad (1)$$

$$d'(t+1) = d'(t)\ \ w(t) + | X - Y| \qquad (2)$$

where

$$| \mathbf{X} - \mathbf{Y} | = \sqrt{\sum_{i=1}^{n} |x_i - y_i|^2} ,$$

|X – Y| measures the difference between neighbor features which indicates groups of features that are similar as well as irrelevant features, d(t+1) and d'(t+1) are winning neurons, d(t) and d'(t) are current neighboring winning neurons, t is number of iterations and w(t) is the weight adjustment of the neuron.

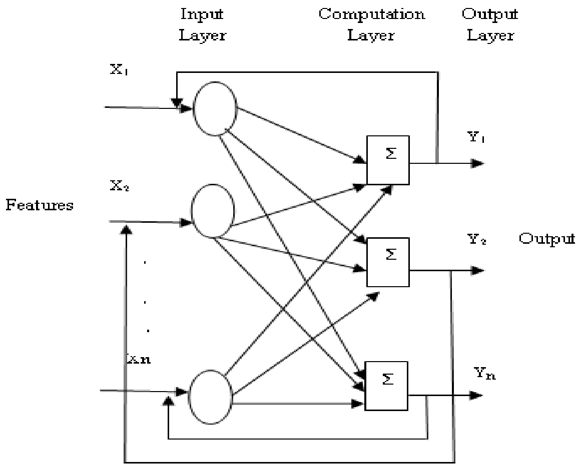

Fig. 2: Structure of Dynamic Self-Organizing Feature Map

Fig. 2 shows the Structure of Dynamic Self-Organizing Feature Map. Input feature vectors of instances $\{X_1,X_2,\ldots, X_n\}$ are fed to the Dynamic Self-Organizing Feature Map. The training of Dynamic Self-Organizing Feature Map follows two phases namely forward phase and reverse phase. Equations (1) and (2) are used in forward phase and reverse phase respectively. In forward phase, the relevant and irrelevant features are identified and grouped. Then irrelevant features are removed and relevant features are sent to the next Feature Associative Memory layer for Feature Interaction.

In the training process, Input feature vectors X can be sub divided into classes which may have similar and dissimilar features. The neurons are connected to adjacent neurons by a neighborhood relation in training process which is iterative. The training process continues until relevant and irrelevant features are classified using (1). After the classification the irrelevant features are filtered and only relevant features are forwarded to the next layer.

In reverse phase, the training process removes noisy data. That is the partially incorrect or partially incomplete features are fully completed through competitive learning method. The reverse phase trains the input vectors X and removes noisy data present in the features using (2). The training consists of choosing a winner neuron by similarity measure and grouping features in the neighborhood of the closest relation. This process is chosen with greatest similarity which is usually defined by means of a distance measure. The computational effort consists of finding a best-matching feature among all the trained neurons. The highest value of feature is the winner of the competition. Finally, the output generated from the reverse phase removes noisy data in the feature. This output of the Dynamic Self-Organizing Feature Map is the input of Feature Associative Memory layer. Hence, Dynamic Self-organizing Feature Map provides relevant features and error free features.

E. Feature Associative Memory:

The Feature Associative Memory component aims to achieve Feature Interaction and removes redundant features. The input of this component is taken from the Dynamic Self-Organizing Feature Map. The structure of the Feature Associative Memory is shown in Fig. 3 and it comprises of recurrent neural network and Cloud Feature Memory database. The recurrent neural network which is implemented employs the concept of Associative Memory. This Associative Memory is used to recall features, based on the degree of similarity between the input patterns (features) and the patterns stored (feature database attributes and dependency) in cloud storage.

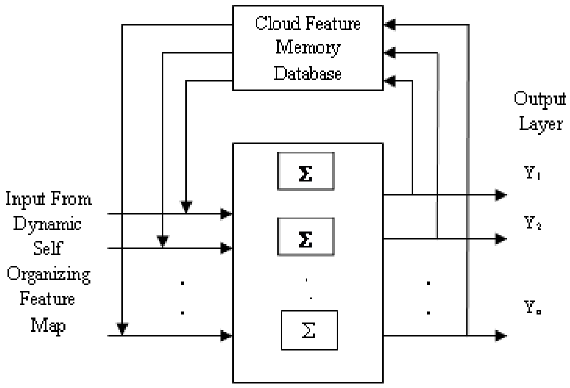

Fig. 3: Structure of Feature Associative Memory

The Cloud Feature Memory database consists of feature database attributes and relationship with the attributes (features). The Feature Associative Memory is used to allow the recall of information based on partial knowledge of its contents. Let X is the input pattern from the Dynamic Self- Organizing Feature Map and Y is the Cloud Feature Memory which contains feature database attributes and dependency. The goal of Feature Associative Memory is to find Feature Interaction without any redundant features. Feature Interactions are the irreducible dependencies between attributes. The recurrent neural network describes the Euclidean distance function to associate certain input patterns with certain memory patterns, i.e., each stored memory consists of a pair of input and desired output patterns. The Feature Interaction between X and Y is defined as follows:

$$d(x,y) = \sqrt{\sum_{i}^{n} (x_i - y_i)^2} \quad 3)$$

where d(x,y) is the square root of the sum of squared differences between corresponding features of the Recurrent neural network and Cloud Feature Memory database.

The Feature Associative Memory computes its output recursively until the network becomes stable. After achieving stable state the actual output is compared with target patterns stored in Cloud Feature Memory database. If the actual output and target pattern matches then the training achieves positive Feature Interaction without any redundant features. Otherwise the training moves to negative Feature Interaction state with less feature dependency.

RESULTS AND DISCUSSION

For experimental purpose we have taken Anneal dataset from Cloud Database AWS (Amazon Web Service) provided by the popular cloud vendor Amazon. The Anneal database consists of 38 attributes which data types are continuous, nominal and integer. For simulation purpose, we have conducted four tests using neural network toolbox in Matlab. Each test contains 1000 sample instances which are taken as input for Dynamic Neural Feature Selection Model. Out of 4000 sample instances, 1500 instances are continuous, 1500 instances are nominal and 1000 instances are integer.

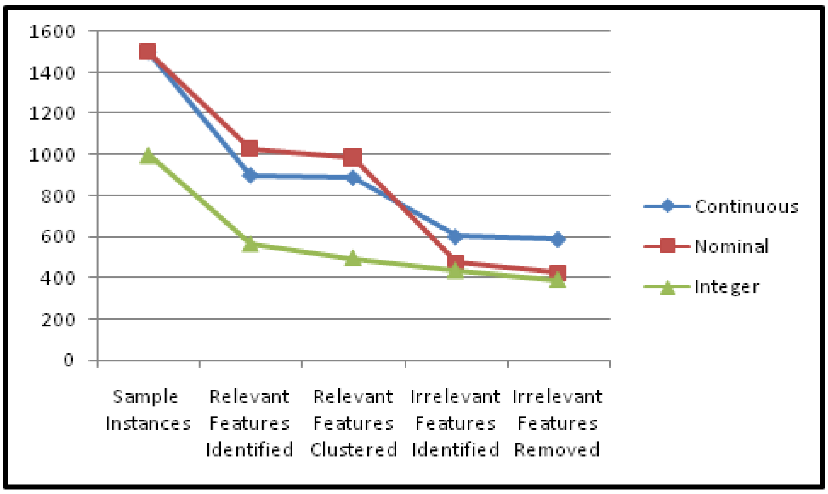

Fig. 4: Forward Phase of Dynamic Self-Organizing Feature Map

Australian Journal of Basic and Applied Sciences, 10(12) July 2016, Pages: 117-125

Dynamic Self-Organizing Feature Map component has forward phase and reverse phase. In forward phase, the network training identifies relevant and irrelevant features and clusters relevant features. Also this phase removes irrelevant features in the experiment. In continuous attributes, 900 relevant features and 600 irrelevant features are identified out of 1500 sample instances. Fig. 4 shows at the end of the forward phase training, 888 features are clustered and 587 irrelevant features are removed. Similarly 1030 nominal relevant features, 565 integer relevant features, 470 nominal irrelevant features, 435 integer irrelevant features are traced out of 2500 sample instances. The sample instances 812 of nominal and integer are eliminated at the end of forward phase. The relevant features 1482 are clustered.

In reverse phase, the neural network training removes noisy in the features using matlab for neural network. The reverse phase trains the 4500 sample instances and 866 features with noisy data are identified. After the completion of training phase, 796 sample instances recovered without any error which is shown in Fig. 5.

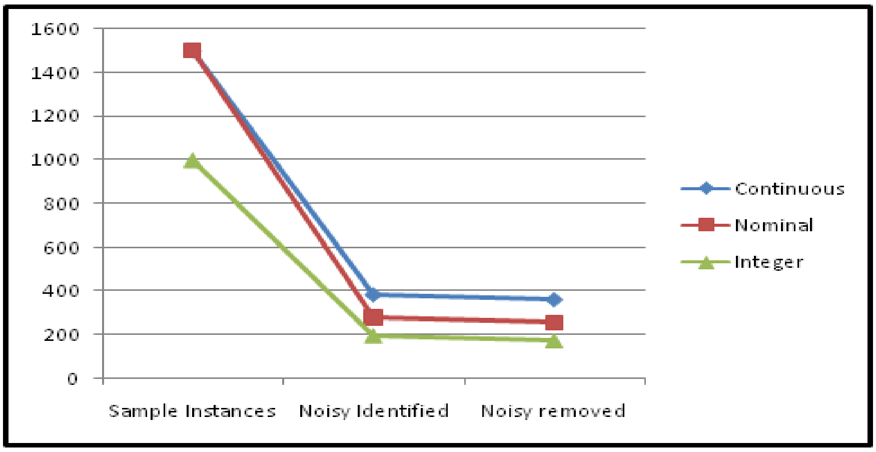

Fig. 5: Reverse Phase of Dynamic Self-Organizing Feature Map

The Feature Associative Memory component achieves Feature Interaction and redundant features. The input 3882 instances are taken from the Dynamic Self-Organizing Feature Map. The whole set of various combination of Anneal dataset are stored in Cloud Feature Memory database. During the training each and every output of recurrent neural network recalls and compares with target output (Cloud Feature Memory database). Fig. 6 shows recall features from the Associative Memory, positive interaction and negative interaction. In the recalling process, 3506 instances are taken for training. The 2303 instances of positive Feature Interactions and 1203 instances of negative Feature Interactions are found during the training of Associative Memory.

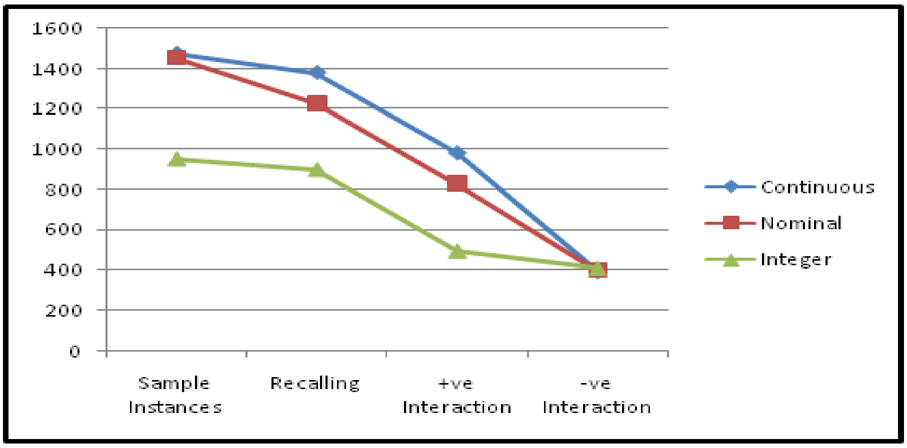

Fig. 6: Feature Interaction in Feature Associative Memory

The performance of "Dynamic Neural Feature Selection Model" is given in Table II. The percentages of relevant features are 98.67%, 95.63% and 87.96% for continuous, nominal and integer type of features respectively. For Continuous, Nominal and integer type of features, the exclusion percentage of irrelevant features are 97.83%, 90% and 89.43% respectively.

Table II: Performance Of Dynamic Neural Feature Selection Model

Category	Relevant Features	Irrelevant Features	Features Without Noisy	Recalling	+ve Interaction	-ve Interaction
Continuous	98.67	97.83	94.03	93.56	71.23	28.77
Nominal	95.63	90.00	91.52	84.31	67.51	32.49
Integer	87.96	89.43	88.38	94.44	54.72	45.28

The forward phase of the training produces better results which are shown in Fig. 7. During the reverse phase of the training the incomplete feature patterns are corrected almost and the percentage of correctness of the patterns are 94.03%, 91.52% and 88.38% for continuous, nominal and integer type of features.

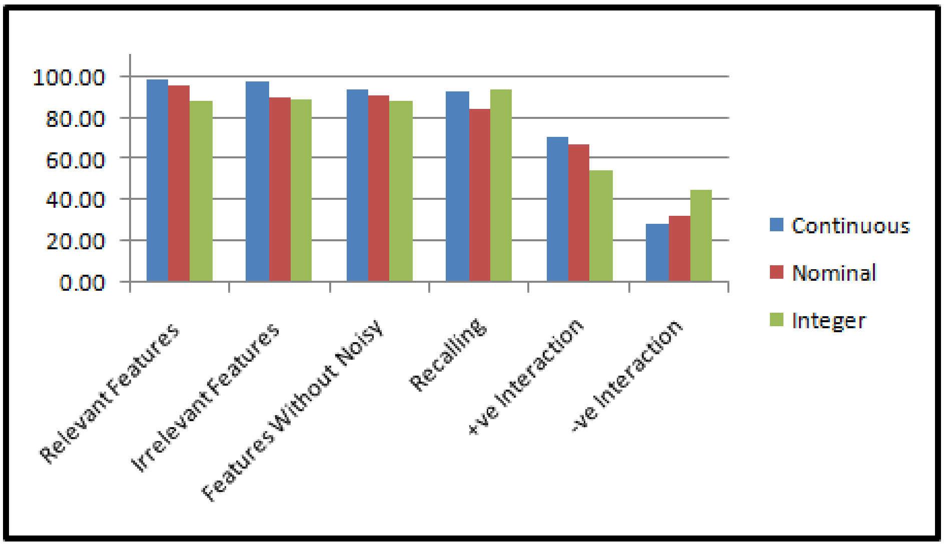

Fig. 7: Performance of Dynamic Neural Feature Selection Model

The Feature Associative Memory is used to recall the features stored in Cloud Feature Memory database which is compared with present output. The Feature Associative Memory calculates the output repeatedly until the network becomes equilibrium state. The percentage of recalling, positive interaction and negative interaction are demonstrated in the Fig. 7 and Table-II. From the above discussion the "Dynamic Neural Feature Selection Model" generates better results than the conventional Feature Selection methods. From the experiment, we shows that the performance of the proposed model is better Feature Selection model for Cloud Databases which supports all kinds of relevant, non-redundant features, features without noisy and Feature Interaction.

Conclusion and Future Directions:

"A Dynamic Neural Feature Selection Model using Cloud Databases" is a new Feature Selection model for Cloud Databases which supports non-redundant, relevant features, Feature Interaction and feature completion. Dynamic Neural Feature Selection Model is compared with the other features selection methods to overcome the limitation of the later. This model comprises Dynamic Self-Organizing Feature Map and Feature Associative Memory. Dynamic Self-Organizing Feature Map is used for relevant features, irrelevant features and feature completion. The Feature Associative Memory is applied for Feature Interaction and redundant features. The result shows that the Dynamic Neural Feature Selection Model is significantly better than the traditional Feature Selection methods. This work can be extended to Hadoop-based neural Feature Selection model and distributed Feature Selection model in Big Data.

REFERENCES

Almuallim, H. and T.G. Dietterich, 1991. Learning with Many Irrelevant Features. In the Proceedings of the 9th National Conference, On Artificial Intelligence, 2: 547-552, Anaheim, CA,. AAAI Press.

Dash, M. and H. Liu, 1998. Hybrid Search of Feature Subsets. In H. Y. Lee and H. Motoda, editors, Proceedings of the 15th Pacific Rim International Conference on AI, pp: 22-27, Singapore, Springer Verlag.

Dea, P.O., J. Griffith and C.O. ‘Riordan, 2001. Combining Feature Selection and neural networks for solving classification problems. In the Proceedings of the 12th Irish Conference on Artificial Intell. Cognitive Sci., pp: 157-166.

ElAlami, M., 2009. A filter model for feature subset selection based on genetic algorithm. Knowledge-Based Systems, 22(5): 356-362.

Fleuret, F., 2004. Fast binary Feature Selection with conditional mutual information. Journal of Machine Learning Research, 5: 1531-1555.

Guyon, I. and A. Elisseeff, 2003. An introduction to variable and Feature Selection. Journal of Machine Learning Research, 3: 1157-1182.

Hodge, V.J., T. Jackson and J. Austin, 2012. A Binary Neural Network Framework for Attribute Selection and Prediction. Paper presented at Proceedings of the 4th International Conference on Neural Computation Theory and Applications (NCTA 2012), Barcelona, Spain, pp: 510-515.

Jakulin, A. and I. Bratko, 2004. Testing the significance of Attribute Interactions. In the Proceedings of Twenty-First International Conference on Machine Learning, pp: 409-416, ACM.

Kira, K., and L. Rendell, 1992. A Practical Approach to Feature Selection. In the Proceedings of the 9th International Conference on Machine Learning, pp: 249-256, Aberdeen, Scotland.

Koller, D., and M. Sahami, 1996. Toward optimal Feature Selection. In Proceedings of International Conference on Machine Learning, pp: 284-292.

Liu, H. and R. Setiono, 1996. A Probabilistic Approach to Feature Selection: a Filter Solution. In Proceedings of the 13th International Conference on Machine Learning, pp: 319-327, Morgan Kaufmann.

Liu, H., H. Motoda and M. Dash, 1998. A Monotonic Measure for Optimal Feature Selection. In Proceedings of the European Conference on Machine Learning, pp: 101-106, Springer Verlag.

Monirul Kabir, M., Md. M. Islam and K. Murase, 2010. A new Wrapper Feature Selection approach using neural network. Journal of Neurocomputing, pp: 3273-3283.

Narendra, P. and K. Fukunaga, 1977. A Branch and Bound Algorithm for Feature Subset Selection. IEEE Transactions on Computer, C–26(9): 917-922.

Peng, H., Fulmi long and C. Ding, 2005. Feature Selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. Pattern Analysis and Machine Intelligence, IEEE: 1226-1238.

Pudil, P., J. Novovicová and J. Kittler, 1994. Floating Search Methods in Feature Selection. Pattern Recognition Letters, 15(11): 1119-1125.

Rajeswari, K., V. Vaithiyanathan and T.R. Neelakandan, 2012. Feature Selection in Ischemic Heart Disease Identification using Feed Forward Neural Networks. In Proceedings of International Symposium on Robotics and Intelligent Sensors, pp: 1818-1823, Elsevier.

Rehman, M., G.M. Khan and S.A. Mahmud, 2014. Foreign Currency Exchange Rates Prediction using CGP and Recurrent Neural Network. In Proceedings of 2014 International Conference on Future Information Engineering, pp: 239-244, Elsevier.

Verikas, A. and M. Bacauskiene, 2002. Feature Selection with neural networks. pp: 1323-1335, Elsevier.

Zhao, Z., and H. Liu, 2009. Searching for interacting features in subset selection. Intelligent Data Analysis, 13(2): 207-228.