ESANN'2004 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 28-30 April 2004, d-side publi., ISBN 2-930307-04-8, pp. 449-458

Neural Networks for Data Mining: Constrains and Open Problems

Răzvan Andonie and Boris Kovalerchuk

Computer Science Department

Central Washington University, Ellensburg, USA

Abstract. When we talk about using neural networks for data mining we have in mind the original data mining scope and challenge. How did neural networks meet this challenge? Can we run neural networks on a dataset with gigabytes of data and millions of records? Can we provide explanations of discovered patterns? How useful that patterns are? How to distinguish useful, interesting patterns automatically? We aim to summarize here the state-of-the-art of the principles beyond using neural models in data mining.

1 What is special in data mining applications?

Data mining (DM) is the nontrivial extraction of implicit, previously unknown, interesting, and potentially useful information (usually in the form of knowledge patterns or models) from data. Historically data mining has grown from large business database applications, such as finding patterns in customer purchasing activities from transactions databases. Original DM problems were to adjust known methods such as decision trees and neural networks (NN) to large datasets (100,000 and more records) and relational database structures. Later methods such as association rules were developed specifically motivated by DM challenge.

The most vehiculated DM problems are reduced to traditional statistical and machine leaning methods: classification, prediction, association rule extraction, and sequence detection. The techniques used in DM are very heterogeneous: statistical methods, case-based reasoning, NN, decision trees, rule induction, Bayesian networks, fuzzy sets, rough sets, genetic algorithms/evolutionary programming.

The following are the major stages in solving a DM problem [7]:

1. Define the problem.
2. Collect and select data, such as deciding which data to collect and how to collect them.

ESANN'2004 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 28-30 April 2004, d-side publi., ISBN 2-930307-04-8, pp. 449-458

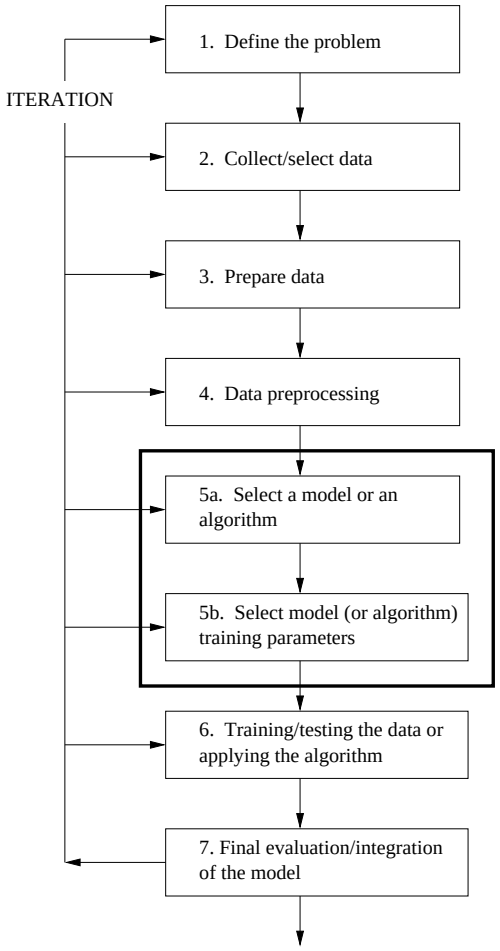

Figure 1: Data modeling process and data mining lifecycle (from [7]).

ESANN'2004 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 28-30 April 2004, d-side publi., ISBN 2-930307-04-8, pp. 449-458

3. Prepare data, such as transform data to a certain format, or data cleansing.
4. Data preprocessing; this task is concerned mainly with enhancement of data quality.
5. Select an appropriate mining method, which consists of:
   (a) Selecting a model or algorithm.
   (b) Selecting model/algorithm training parameters.
6. Training/testing the data or applying the algorithm, where evaluation set of data is used in the trained architecture.
7. Final integration and evaluation of the generated model.

The entire DM process is iterative, and the result in each step can be feed back to any of the previous steps for improvement. The loop will continue until a satisfactory result has been obtained (see Fig. 1).

A lot of work in current DM has been done in developing integrated systems to support all 7 stages not only stages 5 and 6 that are typical for NN and machine learning. These seven steps are well known, but do not include an important step that has recently emerged . This step is "expert mining", as an attempt to integrate patterns derived from data in steps 1-7 with pattern derived from an expert. Such patterns are very valuable if available data are not representative, noise in data is significant and too many trivial patterns are discovered [16].

The first generation of data mining algorithms has been demonstrated to be of significant value across a variety of real world applications, from medicine to homeland security. These first-generation algorithms also have significant limitations [18], and they work best for problems involving a large set of data collected into a single database, where the data are described by numeric or symbolic features [19].

Development of new generation algorithms is expected to encompass more diverse sources and types of data that will support mixed-initiative data mining, where human experts collaborate with the computer to form hypotheses and test them.

The main challenges to the data mining procedure involve the following:

a.) Defining task-specific learning criteria. Traditionally DM inherited from statistics, NN, and machine leaning criteria such as R^2 for numeric target variables and classification errors for nominal target variables. These criteria can be insufficient or even misleading for specific tasks. For instance, in stock market forecasting, two forecasting functions f_1 and f_2 may have the same R^2, but generate different buy/sell signals and different gain/loss in trading.

b.) Massive data sets and high dimensionality.

c.) User interaction and prior knowledge. Data mining is inherently an interactive and iterative process. Users may interact at various stages, and

ESANN'2004 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 28-30 April 2004, d-side publi., ISBN 2-930307-04-8, pp. 449-458

domain knowledge may be used either in the form of a high-level specification of the model, or at a more detailed level. Visualization of the extracted model is also desirable [19].

d.) Overfitting and assessing the statistical significance. Often the presence of spurious data points leads to overfitting of the models.

e.) Understandability of patterns. It is necessary to make the discoveries more understandable to humans.

f.) Noisy, redundant, conflicting, and incomplete data.

g.) Heterogeneous data and mixed media data. We will need algorithms that can learn from multiple databases and the Web, combining numeric and symbolic features with image features, or raw sensor data. It is still largely an unresolved issue of having n heterogeneous attributes of different physical modalities (weights, prices, volumes, etc) combined with a single distance such as the Euclidean distance. Data normalization commonly used in DM only marginally solves this problem because we need to justify normalization coefficients.

h.) Management of changing data and knowledge. Rapidly changing data, in a database that is modified, may make previously discovered patterns invalid.

i.) Integration. It is desirable that DM modules integrate smoothly, both with the database and the final decision making procedure.

j.) Internet applications. DM applications may be related to Internet applications with on-line processing capability, and this requires a short processing time.

k.) Reverse engineering. We will have to develop DM algorithms that go beyond learning to predict likely outcomes, and learn to suggest preemptive actions that achieve the desired outcome [18]. We call this feature "reverse engineering". After predicting an outcome, we should be able to find the most influential factors that have caused this prediction. Going from the effect to the cause is the way we can optimize decisions, rather than predictions.

l.) Biased samples of data. We have to answer the following question: How can a system learn from biased samples of data? The difficult issue is that the available data often represents a biased sample that does not correctly represent the underlying causes and effects [18]. This question is related to another one: How do we select the learning data?

m.) Optimal generation of experiments. Most current DM systems are tested using predetermined data sets (e.g., from public repositories such as the UCI Machine Learning repository). We need algorithms that actively generate optimal experiments for each DM problem [18].

These are hard requirements and the question is how far we can go with using NN for DM applications.

ESANN'2004 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 28-30 April 2004, d-side publi., ISBN 2-930307-04-8, pp. 449-458

2 Neural networks for data mining

We aim to summarize here the state-of-the-art of the principles beyond using neural models in data mining, and not of the applications. A non-technical book on NN for data mining is [4]. For NN people, more consistent material can be found in the IEEE Trans. on NN Special Issue (June 2000), and in the survey of Mitra et al. [19]. Almost all data mining books have chapters on NN, but only few of them are more then introductive.

Neural networks are suitable in data-rich environments and are typically used for extracting embedded knowledge in the form of rules, quantitative evaluation of these rules, clustering, self-organization, classification and regression, feature evaluation and dimensionality reduction.

In a 1999 survey of 43 DM software products, which were either research prototypes or commercially available, Goebel and Gruenwald [10] found 10 NN related products: BrainMaker, Clementine, Intelligent Miner (IBM), Darwin, Data Surveyor, Decision Series, Kepler, Delta Miner, ModelQuest, and ToolDiag.

Many standard software packages for data mining contain neural network modules. However, some of these modules are extremely basic: most of the time just a simple multi-layered perceptron, trainable with inefficient and old-fashioned updating techniques such as standard Backpropagation. They often fail to fulfil the important requirement of providing insight in the database. In fact, one could even argue whether these standard NN are truly methods for data mining as defined above, or at most classification, predictions and perhaps clustering tools.

The IEEE Neural Networks Society is on the way to become a Computational Intelligence Society and this reflects the trend to integrate neural computation into hybrid methods also known as soft computing tools. Soft computing is a consortium of methodologies that works synergistically and provides, in one form or another, flexible information processing capability for handling real-life ambiguous situations. Its aim is to exploit the tolerance for imprecision, uncertainty, approximate reasoning, and partial truth in order to achieve tractability, robustness, and low-cost solutions [19].

Soft computing methodologies (involving fuzzy sets, NN, genetic algorithms, and rough sets) are most widely applied in the DM. Fuzzy sets provide a natural framework for the process in dealing with uncertainty. NN and rough sets are used for classification and rule generation. Genetic algorithms are involved in various optimization and search processes, like query optimization and template selection. It is presently hard to separate NN as a distinct tool. For instance, some of the most efficient soft computing rule generation methods are neuro-fuzzy systems [2].

The latest developments in research on NN bring them much closer to the ideal of data mining: knowledge out of data in understandable terms. Methods have been developed for the simplification ("pruning") and visualization of NN, for input relevance determination, and to discover symbolic rules out of trained NN.

ESANN'2004 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 28-30 April 2004, d-side publi., ISBN 2-930307-04-8, pp. 449-458

NN and their soft computing hybridizations have been used in a variety of DM tasks [3]. We can say that the main contribution of NN toward DM stems from rule extraction and from clustering.

Rule Extraction and Evaluation: Typically a network is first trained to achieve the required accuracy rate. Redundant connections of the network are then removed using a pruning algorithm. The link weights and activation values of the hidden units in the network are analyzed, and classification rules are generated [23]. The generated rules can be evaluated according to some quantitative measures (e.g., accuracy, coverage, fidelity, and confusion). This relates to the preference criteria/goodness of fit chosen for the rules. It seems that from the global DM perspective rules extraction from NN is a temporary solution for getting interpretable results. Direct rule extraction from data potentially can produce better rules. Extraction rules from NN may carry NN limitations and artifacts to rules.

Clustering and Dimensionality Reduction: Kohonen's SOM [14] proved to be an appropriate tool for handling huge data bases. Kohonen et al. [15] have demonstrated the utility of a SOM with more than one million nodes to partition a little less than seven million patent abstracts where the documents are represented by 500-dimensional feature vectors. Kohonen's LVQ [13] was successfully used for on-line dimensionality reduction [12], [6]. SOM and LVQ, used with data visualization techniques, are presently one of the most promising NN application in DM. The main reason for this is the scalability of the SOM model. Meanwhile, dimensionality reduction is essential for data visualization and analysis.

Incremental Learning: When designing and implementing data mining applications for large data sets, we face processing time and memory space problems. In this case, incremental learning is a very attractive feature. The fundamental issue in incremental learning is: how can a learning system adapt to new information without corrupting or forgetting previously learned information – the so-called *stability-plasticity* dilemma addressed by Carpenter and Grossberg [5]. In the context of supervised training, incremental learning means learning each input-output sample pair, without keeping it for subsequent processing. Very few algorithms perfectly fit into this description of incremental learning. The fuzzy ARTMAP (FAM) family of neural networks [1] is the best known example. The FAM model has been incorporated in the MIT Lincoln Lab system for data mining of geospatial images because of its computational capabilities for incremental learning, fast stable learning, and visualization [21].

We are analyzing here the role of NN in DM. However, the field of NN is itself about to undergo a further change in orientation and self-conception, entering "The Second Start–Up". NN of today is at the bifurcation to two possible future alternatives: *Innovate!* or *Vanish!* [11].

The key question addressed by this special session on "Neural Networks for Data Mining" is: How can we bridge the gap between the state-of-the-art in neural network research for data mining and the NN implemented in standard data mining software? Some of the main challenges raised by DM applications

ESANN'2004 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 28-30 April 2004, d-side publi., ISBN 2-930307-04-8, pp. 449-458

are addressed by the papers of the session are: *i)* Dealing with high-dimensional data, *ii)* Clustering and Self Organizing Maps, *iii)* Visualization, and *iv)* Applications. Other aspects, like rule extraction, are not discussed here.

3 Neural networks are great, but...

There are many nice features of NN, which make them attractive for DM. These features include learning and generalization ability, adaptivity, content addressability, fault tolerance, self-organization, robustness, and simplicity of basic computations [8]. How far can we go with our neural models in data mining without doing data inquisition where "the data are tortured until they confess" [3] ?

NN are known to be especially useful for problems characterized by:

- A large amount of example data is available and it is difficult to specify a parametric model for the data.
- High input dimension and relationships exist within the data that are not fully understood (black box).
- There are potentially stable patterns in the data that are subtle or deeply hidden.
- The data exhibit significant uncharacterizable nonlinearity.
- Iterative use of the data is required to detect patterns.
- Problems are solved by generating predictions of complicated phenomena rather than by generating explanations.

For many, the general impression is (or was) that NN are not necessarily a natural choice for DM. At the level of the year 1996, the major criticism was [17]:

- NN learn by many passes over the training set so that the learning time of NN is usually long.
- A NN can not expose its knowledge as symbolic rules.
- Available domain knowledge is rather difficult to be incorporated to a NN.

How different are things now? During the last years, NN have evaluated significantly and we have partial answers to these critiques. NN are now extracting symbolic rules and can learn relatively fast.

A fundamental critique is that there is no general theory that specifies the type of neural network, its architecture, or learning algorithm for a given problem. As such, network builder must experiment with a large number of NN before converging upon the appropriate one for the problem in hand. According

ESANN'2004 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 28-30 April 2004, d-side publi., ISBN 2-930307-04-8, pp. 449-458

to Roy [22], a weakness in the theories of artificial NN is the total absence of the concept of an *autonomous learning algorithm*. The learning algorithms need constant "baby-sitting" in order for them to work - learning rates need to be reset and readjusted, various network designs need to be tried so that they can generalize well, starting points need to be reset when they get stuck in a local minimum, and everything needs to be re-learned from scratch when there is catastrophic forgetting in the network. These are some of the common problems in both supervised and unsupervised neural network learning.

We are not going to analyze Roy's call for a "shake up of the field of neural networks" [22]. Our only observation is that this "baby-sitting" is very similar to the iterative stages in solving a DM problem using other methods. Using a NN does not mean to solve completely automatically a DM problem. We still need to re-iterate these stages, choosing between the different neural models and fitting their parameters.

On the other hand, NN have advantages, over other types of machine learning algorithms, for scaling [3] and learning [9]. A comparison of NN models and problem requirements for a stock market prediction problem is described in [16].

4 Conclusions

Compared to statistical methods, NN are useful especially when there is no a priori knowledge about the analyzed data. They offer a powerful and distributed computing architecture, with significant learning abilities and they are able to represent highly nonlinear and multivariable relationships [20]. However, NN are not appropriated for any DM problem and the selection of a network architecture for a specific problem has to be done carefully.

We have not attempted to provide an exhaustive survey of the available NN algorithms that are suitable for DM. Instead, we have described a subset of the problems and constrains, selected to illustrate the breath of relevant approaches as well as the key issues that arise in applying NN in a DM setting.

References

[1] Carpenter G. A., Grossberg S., Markuzon N., Reynolds J. H., and Rosen D. B. Fuzzy ARTMAP: A neural network architecture for incremental supervised learning of analog multidimensional maps. *IEEE Trans. Neural Networks*, 3:698–713, 1992.

[2] S. Abe. *Pattern Classification: Neuro-Fuzzy Methods and Their Comparison.* Springer Verlag, London, 2001.

[3] Y. Bengio, J. M. Buhmann, M. Embrechts, and J. M. Zurada. Introduction to the special issue on neural networks for data mining and knowledge discovery. *IEEE Trans. Neural Networks*, 11:545–549, 2000.

ESANN'2004 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 28-30 April 2004, d-side publi., ISBN 2-930307-04-8, pp. 449-458

[4] J. P. Bigus. *Data Mining with Neural Networks: Solving Business Problems - from Application Development to Decision Support*. McGraw-Hill, New York, 1996.

[5] G.A. Carpenter and S. Grossberg. The ART of adaptive pattern recognition by a self-organizing neural network. *IEEE Computer*, 21:77–88, 1988.

[6] A. Cataron and R. Andonie. RLVQ determination using OWA operators. In M. Hamza, editor, *Proceedings of the Third IASTED International Conference on Artificial Intelligence and Applications (AIA 2003), Benalmadena, Spain, September 8-10*, pages 434–438, ACTA Press, 2003.

[7] Z. Chen. *Data Mining and Uncertain Reasoning: An Integrated Approach*. Wiley, 2001.

[8] K. J. Cios, W. Pedrycz, and R. Swiniarski. *Data Mining Methods for Knowledge Discovery*. Kluwer, Boston, 1998.

[9] M. W. Craven and J. W. Shavlik. Using neural networks for data mining. *Future Generation Computer Systems*, 13:211–229, 1997.

[10] M. Goebel and L. Gruenwald. A survey of data mining and knowledge discovery software tools. *ACM SIGKDD Explorations*, 1:20–33, 1999.

[11] N. Goerke. Quo vadis neurocomputing? neural computation at the edge to new perspectives. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN 2003)*, pages 660–665, Portland, Oregon, July 20-24 2003.

[12] B. Hammer and T. Villmann. Generalized relevance learning vector quantization. *Neural Networks*, 15:1059–1068, 2002.

[13] T. Kohonen. Improved versions of learning vector quantization. In *Proc. Int. Joint Conf. on Neural Networks*, pages 545–550, San Diego, 1990.

[14] T. Kohonen. *Self-Organizing Maps*. Springer Verlag, 1997.

[15] T. Kohonen, S. Kaski, K. Lagus, J. Salojarvi, J. Honkela, V. Paatero, and A. Saarela. Self organization of a massive document collection. *IEEE Trans. Neural Networks*, 11:574–585, 2000.

[16] B. Kovalerchuk and E. Vityaev. *Data Mining in Finance: Advances in Relational and Hybrid Methods*. Kluwer Academic Publishers, 2000.

[17] H. Lu, R. Setiono, and H. Liu. Effective data mining using neural networks. *IEEE Trans. Knowl. Data Eng.*, 8(6):957–961, 1996.

[18] T. Mitchell. Machine learning and data mining. *Communications of the ACM*, 42(11):30–36, 1999.

ESANN'2004 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 28-30 April 2004, d-side publi., ISBN 2-930307-04-8, pp. 449-458

[19] S. Mitra, Pal S. K., and Mitra P. Data mining in soft computing framework: a survey. *IEEE Trans. Neural Networks*, 13(1):3–14, 2002.

[20] A. Nürnberger, W. Pedrycz, and R. Kruse. Neural network approaches. In W. Klösgen and J. M. Żytkow, editors, *Handbook of Data Mining and Knowledge Discovery*, pages 304–317. Oxford University Press, 2002.

[21] O. Parsons and G. A. Carpenter. ARTMAP neural network for information fusion and data mining: map production and target recognition methodologies. *Neural Networks*, 16:1075–1089, 2003.

[22] A. Roy. Artificial neural networks - a science in trouble. *ACM SIGKDD Explorations*, 1:33–38, 2000.

[23] A. B. Tickle, R. Andrews, M. Golea, and J. Diederich. The truth will come to light: Directions and challenges in extracting the knowledge embedded within trained artificial networks. *IEEE Trans. Neural Networks*, 9(5):1057–1068, 1998.

Neural networks have been successfully applied in a wide range of supervised and unsupervised learning applications. Neural-network methods are not commonly used for data-mining tasks, however, because they often produce incomprehensible models and require long training times. In this article, we describe neural-network learning algorithms that are able to produce comprehensible models, and that do not require excessive training times. Neural networks are supposed toâ€¦Â Neural networks are supposed to be able to mimic any continuous function. But many a times we are stuck with networks not performing up to the mark, or it takes a whole lot of time to get decent results. One should approach the problem statistically rather than going with gut feelings regarding the changes which should be brought about in the architecture of the network. One of the first steps should be proper preprocessing of data. Other than mean normalisation and scaling, Principal Component Analysis may be useful in speeding up training. If the dimension of the data is reduced to such an e