International Journal of Advanced Research in

Computer Science Engineering and Information Technology

Volume: 3 Issue: 3 11-Nov-2014,ISSN_NO: 2321-3337

A PERFORMANCE OF MACHINE LEARNING ALGORITHM

J.SHARMILA[1] MCA.,M.phil., DR.A.SUBRAMANI[2]

Research Scholar, Manonmaniam Sundaranar University, Tirunelveli, Tamilnadu, India.[1]

Professor & Head, Department of Computer Applications, K.S.R. College of Engineering, Thiruchengode, Tamilnadu, India.[2]

ABSTRACT−*Machine learning, a branch of Artificial Intelligence is about the construction and study of systems that can learn from data. It focuses on prediction, based on known properties learned from the training data. Data mining focuses on the discovery of unknown properties on the data. The Machine learning also employs data mining methods as unsupervised learning or as a preprocessing step to improve learner accuracy. The performance of Machine Learning Algorithm is usually evaluated with respect to the ability to reproduce known knowledge, while in Knowledge Discovery and Data Mining (KDD) the key task is the discovery of previously unknown knowledge. Evaluated with respect to known knowledge, an uninformed (unsupervised) method will easily be outperformed by supervised methods, while in a typical KDD task, supervised methods cannot be used due to the unavailability of training data...*

Keywords: Machine Learning Algorithm – Data Mining – Learning.

1, INTRODUCTION

This paper presents a novel framework for solving the wrapper adaptation with new attribute discovery via Machine Learning. One objective of our approach is to automatically adapt a previously learned wrapper from a source Web site to a new unseen site. The second objective is to tackle the problem of new attribute discovery which is to find new attributes that are not specified in the learned or adapted wrapper. It is also able to discover semantic labels for the new attributes discovered. A semantic label refers to the text fragment on the Web page indicating the name of the attribute.

There are five types of machine learning algorithms are as follows:

1. Supervised learning: Generates a function that maps inputs to desired outputs (also called **labels**, because they are often provided by human experts labelling the training examples).

2. **Unsupervised learning:** Find natural classes for examples models a set of inputs, like clustering labels are not known during training.

3. Semi-supervised learning: combines both labelled and unlabeled examples to generate an appropriate function

4. Reinforcement learning: Learns how to act given an observation of the world. Every action has some impact in the environment, and the environment provides feedback in the form of rewards that guides the learning algorithm.

5. Learning to learn: Learns its own inductive bias based on previous training data set.

2, Related Review

Ashraf F *et al* have proposed a system, where clustering techniques have been used for automatic IE from HTML documents having semi-structured data. By means of domain-specific information provided by the user, the proposed system has parsed and tokenized the data from an HTML document, divided it into clusters having analogous elements, and estimated an extraction rule based on the pattern of occurrence of data tokens. Then, the extraction rule has been utilized to refine clusters, and finally, the output has been demonstrated. Moreover, a multi-objective genetic algorithm-based clustering method has been used for finding the number of clusters and the most natural clustering. It is complex and even impossible to employ a manual approach to mine the data records from web pages in deep web. Thus, *Chen Hong-ping et al* [9] have proposed a LBDRF algorithm to solve the problem of automatic data records extraction from Web pages in deep Web. Experimental result has shown that the proposed technique has performed well.

Deep architectures are composed of multiple levels of non-linear operations, such as in neural nets with many hidden layers or in com- plicated propositional formulae re-using many sub-formulae. Searching the parameter space of deep architectures is a difficult task, but learning algorithms such as those for Deep Belief Networks have recently been proposed to tackle this problem with notable success, beating the state- of-the-art in certain areas. This monograph discusses the motivations and principles regarding learning algorithms for deep architectures, in particular those exploiting as building blocks unsupervised learning of single-layer models such as Restricted Boltzmann Machines, used to construct deeper models such as Deep Belief Networks.

3, Learning Algorithm

There are three types of learning problem such as,

1.Learning Problem A learning problem is defined by probability distribution $D(x,y)$ over features x which are a vector of bits and a label y which is either 0 or 1.

2. **Shallow Learning Problem** A shallow learning problem is a learning problem where the label y can be predicted with error rate at most $e < 0.5$ by a weighted linear combination of features, $sign(sum_i\ w_i\ x_i)$.

3. Deep Learning Problem A deep learning problem is a learning problem with a solution represent able by a circuit of weighted linear sums with O(number of input features) gates.

The problem is parameterized by an integer k, where larger k problems hold for smaller choices of e. An example is drawn by first picking a uniform random bit y from $\{0,1\}$. After that k hidden bits $h_1,\ldots,h_k$ are set so that a random subset of $(k + y)/2$ of them are 1 and the rest 0. For each hidden bit h_i, we have 4 output bits x_{i1}, $xi2,xi3,xi4$ (implying a total of $4k$ output bits). If $h_i = 0$, with 0.5 probability we set one of the output bits to 1 and the rest to 0, and with 0.5 probability we set all output bits to 0. If $h_i = 1$, with 0.5 probability we set one of the output bits to 0 and the rest to 1, and with 0.5 probability we set all output bits to 1.

Variations using recursive composition (redefine each "output bit" to be a hidden bit in a new layer, each of which has its own output bit) can make the "right" number of levels be larger than 2.

Builds a threshold weighted sum predictor for every feature x_{ij} using weights = the probability of agreement between the features minus 0.5.

Builds a threshold weighted sum predictor for the label given the predicted values from the first step with weights as before.

For each output feature x_{ij}, the values of output features corresponding to other hidden bits are uncorrelated since by construction $Pr(h_i = h_{i'}) = 0.5$ for $i \mathrel{!}= i'$.

For output features which share a hidden bit, the probability of agreement in value between two bits j,j' is 0.75.

If we have n IID samples from the learning problem, then Cher off bounds imply that empirical expectations deviate from expectations at most $(\log ((4k)^2/d)/2n)^{0.5}$ with probability d or less for all pairs of features simultaneously.

Learning Algorithm are specified,

A model: The information processing unit of the deep learning

an architecture: A set of web pages and links connecting web page. Each link has a weight,

a learning algorithm: used for training the web pages by modifying the weights in order to model a particular learning task correctly on the training examples.

Learning Process For Perceptron,

Initially assign random weights to inputs between -0.5 and +0.5

Training data is presented to perceptron and its output is observed.

If output is incorrect, the weights are adjusted accordingly using following formula.

$$wi \leftarrow wi + (a^* xi^* e), \text{ where 'e' is error produced}$$

and 'a' $(-1 < a < 1)$ is learning rate

'a' is defined as 0 if output is correct, it is +ve, if output is too low and –ve, if output is too high.

Once the modification to weights has taken place, the next piece of training data is used in the same way.

Once all the training data have been applied, the process starts again until all the weights are correct and all errors are zero.

Each iteration of this process is known as an epoch.

CONCLUSION

Automated TC is now a major research area within the information systems discipline, thanks to a number of factors

—Its domains of application are numerous and important, and given the proliferation of documents in digital form they are bound to increase dramatically in both number and importance.

—It is indispensable in many applications in which the sheer number of the documents to be classified and the short response time required by the application make the manual alternative implausible.

—It can improve the productivity of human classifiers in applications in which no

Classification decision can be taken without a final human judgment..

—It has reached effectiveness levels comparable to those of trained professionals.

The effectiveness of manual TC is not 100% anyway [Cleverdon 1984] and, more importantly, it is unlikely to be improved substantially by the progress of research. The levels of effectiveness of automated TC are instead growing at a steady pace, and even if they will likely reach a plateau well below the 100% level, this plateau will probably be higher that the effectiveness levels of manual TC.

References

Amati, G. and Crestani, F. 1999. Probabilistic learning for selective dissemination of information. Information Processing and Management 35, 5, 633–654.

Androutsopoulos, I., Koutsias, J., Chandrinos, K. V., and Spyropoulos, C. D. 2000.An experimental comparison of naive Bayesian and keyword-based anti-spam filtering with personal e-mail messages. In Proceedings of SIGIR-00, 23rd ACM International Conference on Research and Development in Information Retrieval (Athens, GR, 2000), pp. 160–167.

Apt´e, C., Damerau, F. J., and Weiss, S. M. 1994. Automated learning of decision rules for text categorization. ACM Transactions on Information Systems 12, 3, 233–251.

Attardi, G., Di Marco, S., and Salvi, D. 1998. Categorization by context. Journal of Universal Computer Science 4, 9, 719–736.

Baker, L. D. and McCallum, A. K. 1998. Distributional clustering of words for text classification. In Proceedings of SIGIR-98, 21st ACM International Conference on Research and Development in Information Retrieval (Melbourne, AU, 1998), pp. 96–103.

Belkin, N. J. and Croft, W. B. 1992. Information filtering and information retrieval: two sides of the same coin? Communications of the ACM 35, 12, 29–38.

Biebricher, P., Fuhr, N., Knorz, G., Lustig, G., and Schwantner, M. 1988. The automatic indexing system AIR/PHYS. From research to application. In Proceedings of SIGIR-88, 11th ACM International Conference on Research and Development in Information Retrieval (Grenoble, FR, 1988), pp. 333–342. Also reprinted in [Sparck Jones and Willett 1997], pp. 513–517.

Borko, H. and Bernick, M. 1963. Automatic document classification. Journal of the Association for Computing Machinery 10, 2, 151–161.

Caropreso, M. F., Matwin, S., and Sebastiani, F. 2001. A learner-independent evaluation of the usefulness of statistical phrases for automated text categorization. In A. G. Chin Ed., Text Databases and Document Management: Theory and Practice. Hershey, US: Idea Group Publishing. Forthcoming.

BIOGRAPHY

Prof. J.sharmila received MCA, M.Phil from Bharathidasan University, Trichy, Tamilnadu, India in the year of 2000 and 2004. He is currently pursing his Ph.D in Manonmaniam Sundaranar University, Tirunelveli, Tamilnadu, India. At Present working as an Assistant Professor in Department of Computer Applications, Bharathidasan UniversityConstituent College(W), Orathanadu, Thanjavur Dt, Tamil Nadu, India. Her Research interested includes Data Mining.

Dr. A.Subramani received his Ph.D Degree in Computer Applications from Anna University, Chennai. He is now working as a Professor & Head, Department of Computer Applications, K.S.R. College of Engineering, Thiruchengode, Tamilnadu, India. His research interested includes ATM Networks, Ad Hoc Networks, High Speed Networks. He has published more that 28 technical papers at various National / International Conference and Journals. He is a life member of ISTE, CSI.