Optimizing Feature Relevance: A Deep Learning Perspective

Gaddam Kavyasri

Lead-Technology, Synechron Technologies Pvt Ltd, Taramani, Chennai-600113, Tamilnadu, India

Abstract

Feature Selection has turned into the main point of investigations particularly in bioinformatics where there are numerous applications. Deep learning technology is a useful asset to choose features; anyway, not all calculations are on an equivalent balance with regards to the selection of relevant features. To be sure, numerous techniques have been proposed to select multiple features using deep learning techniques. Because of deep learning, neural systems have profited a gigantic toprecovery in the previous couple of years. Anyway, neural systems are black-box models and not many endeavors have been made to examine the fundamental procedure. In this proposed work a new calculation to do feature selection with deep learning systems is introduced. To evaluate our outcomes, we create relapse and grouping issues that enable us to think about every calculation on various fronts: exhibitions, calculation time, and limitations. The outcomes acquired are truly encouraging since we figure out how to accomplish our objective by outperforming irregular backwoods exhibitions for each situation. The results prove that the proposed method exhibits better performance than the traditional methods.

Keywords: Feature Selection, Deep Learning, Neural Networks, Preprocessing, Data Extraction

1. Introduction

Variable and feature selection became the main target of much research, especially in bioinformatics where there are many applications. Machine learning may be a powerful tool to choose features, however not all machine learning algorithms are on an equal footing when it involves feature selection [1]. Indeed, many methods are proposed to grab out feature selection with random forests, which makes them the present go-to model in bioinformatics[2]. This mainly comes from the fact thatrandom forests are well known to be good out-of-the-bag algorithms and they donot need a huge amount of data to achieve good results[3].On the other hand, thanks to the so-called deep learning, neural networks have benefited huge interest resurgence in the past few years.

However neural networks are black-box models and really few attempts are madeto research the underlying process [4-9]. Indeed, quite a few articles are often found about feature extraction with neural networks (for which the underlying inputs-outputs process doesn't get to be understood), while only a few tackle feature selection [10-15]. Furthermore,neural networks are known to require lots of data and computation time toachieve good performances. Since data are often hard to obtain in bioinformatics,this is already a burden for neural networks. Nevertheless, some attempts weremade to select features using neural networks, unfortunately, most of them used veryshallow networks and others were directed to very specific datasets [16-20].

Consider a binary classification problemwith a class corresponding to a positive outcome (for example an alarm activation)and the other to a negative outcome (the alarm doesn't activate). Also, consider a binary

classification model that is used to classify an input (for example a motiondetector) to one of the two classes [21-27]. For each data sample, the classification madeby the model belongs to one of the following categories:

True positive (TP): This occurs if the model activates the alarm when itshould have been. No error is made.

True negative (TN): This occurs if the model doesn't activate the alarmrightfully so (i.e. the alarm should not have been activated). No error ismade.

False-positive (FP): This occurs when the model activates the alarm although it should not have been. An error is made and leads to a Type 1 error.

False-negative (FN): This occurs when the model doesn't activate the alarmalthough it should have been. An error is made and leads to a Type 2 error.

A neural network can be built in a plentitude of ways and are subject to manyparameters, neural architecture being the first one . Indeed, neural networks cantake many forms, ranging from very shallow to very deep and very narrow to verywide. Many constraints can also be added in the architecture itself, convolutionaland encoder layers are some of them. All of these parameters can be changedregarding the problem we are facing . In our case, we decided to limit ourselves totest our algorithms on networks with fully connected hidden layers. Wedid this choice since our data didn't give us a priori reason to introduce structureinto our network. Furthermore, this is the more generic and \simplest" architecturethat can be found.

Dropout can be seen as an \ensemble" method for neural networks.Indeed, the principle is to train only a subpart of the network at each iteration.Each neuron has a given probability to be temporarily removed" at traintime. At test time, all neurons are used and their weights are adapted regardingtheir probability of being kept at training time. This can be seen as training multiplenetworks and averaging their predictions at test time (although this is not reallywhat happens, it would be too costly to train multiple networks).

2. Literature Survey

Li, Y et al [1] proposed multiple formulae to carry out feature selection. They were separated into three categories: zero order, first order, and second-order methods. They were directly based on the parameters of the network while first and second-ordermethods were respectively based on the derivative and second derivatives ofthose parameters.

Marbach, D et al [2] proposed to use one of the formulae mentionedinto tackle deeper neural networks. Indeed, a back-propagation method is used to compute feature importance. Let i be the neuron whose importance score we are calculating,and Ni the set of neurons in the next layer (closer to output) that i feeds into.

Montavon et al [3] gave some insight into how toassociate neural activation and feature importances. The idea here consistsof analyzing the activation of the neurons for each input sample and averaging over all samples, thus using each data sample values and not basing theformula only on the network's intern parameters. This technique is proposedand works as follows. Let xi be the ith dimensionof the input example x connected to jth hidden neuron by w_{ji} and b_j the biasof hidden neuron j.

Unfortunately, the regularization method we used doesn't allow us to select redundant features. Indeed, imagine we have two features representing the same information. Since the regularization is linear, it is equivalent cost-wise to have one big andone small weight rather than two medium one (note that this is beneficial whenone wants to select as few variables as possible, i.e. to solve the minimal optimalproblem. To counter this (i.e. to solve the all-relevantproblem), quadratic regularization (elastic net) could be introduced and wouldhelp the network selecting both of the variables to minimize the cost.

3. Proposed Method

The goal of this subsection is to give the formula to compute importance definedas how much a given neuron contributes to the output \variability", to this end wewill go through the 5 following parts:

1. Importance metric definition. This paragraph will define the importancemeasure of each neuron for a given data sample. This measure is based onneural activation[54-56]. However, they only analyze thecontribution of the inputs on the first hidden layer, whereas here we proposea formula that takes the whole network structure into account.

2. Initialization. In this paragraph, a method for initializing the algorithm willbe discussed. We will also give some clues on how this technique could be refined in different settings[57].

3. Backpropagation. We will explain how the two first parts are put togetherto obtain the algorithm.
4. Results. We will show an example of the importance that are obtained usingthis method and show that the results seem reasonable.
5. Extensions. In this sub subsection we will show that given the results obtained, this method might also be used to prune neural networkswithout hurting accuracy.

Only intern parameters are used with that method,whereas ours also uses data samples to compute neural activation. Our algorithm is presented hereunder:

Algorithm III. A general algorithm for back-propagation feature selection methods.

1. Train a network (or use a pre-trained network).
2. For each training sample, do the following steps :
   (a) Initialization phase: Assign importance to the neurons of the network's last layer, by propagating the training sample through the network.
   (b) Back-propagation (step one): Use the importance of neurons fromlayer i to compute those of layer i_1, where layer i is the one for whichimportance has already been assigned and is the furthest away fromthe output.
   (c) Back-propagation (step two): Repeat step (b) until importancehas been assigned to the input layer's neurons.
   (d) Store importance: The features importanceof this input samplecorrespond to the neurons' importance of the input layer and need to bestored.
3. Repeat step (2) for each training sample and sum all the feature importance.
4. The sum finally obtained corresponds to the overall feature importances.

In a single output regression problem setting, we also need to consider negativeoutput values the same way as positive ones. This leads to the following initialization (with w_i the weights connecting the last hidden layer to the output) :

$$Imp(n_i) = |Out(n_i) * w_i|$$

In multi-output regression settings, we make the hypothesis that each outputneuron has the same importance. This way, if we let n_1; :::; n_k be the neurons ofthe last hidden layer and m1; :::;ml be the output neurons (i = 1 if it is a singleoutput problem). We have (with w_{ix} the weights connecting the last hidden layerto output x):

$$Imp(n_i) = \sum_{x=1}^{l} |Out(n_i) * w_{ix}|$$

In classification problems, softmax layers are often added before the output layersuch that the output neurons correspond to probabilities of belonging to a givenclass. Softmax layers map an N-dimensional vector v of arbitrary real values toanother N-dimensional vector soft(v) with values in the range [0; :::; 1] that add upto 1. This is done by using the following formula:

$$soft(v)_j = \frac{e^{v_j}}{\sum_{n=1}^{N} e^{v_n}}, \text{ for j} = 1,...,\text{n}$$

This method might not be optimalsince we consider each neuron of the softmax layer as equally important, and wedo not consider negative and positive values differently. For example, for a binaryclassification problem, we make no difference if the inputs of the softmax layer are[0:7; 0:2] or [0:7;0:2].

4. Results and Discussions

We are now going to look at the results for a regression problem. The dataset wewill use has 5000 input features [x1; :::; x5000]. The regression problem has been generated using the following formula, where the weights have been chosen uniformlyat random between 0 and 100:

$$y = \sum_{i=1}^{25} w_i * x_i$$

This means that only 25 out of the 5000 input features are useful to predictthe output. Figure 1 shows the importance of each neuron (layer by layer). Inthis case, the neural network used has 4 hidden layers of 500 neurons each. Layer 0 is the input layer and thus corresponds to the featureimportance. As we can see, there is peak importance on the first neurons oflayer 0, this is expected (due to the problem nature, which only uses the 25 firstfeatures) and proves that the technique used seems reasonable.

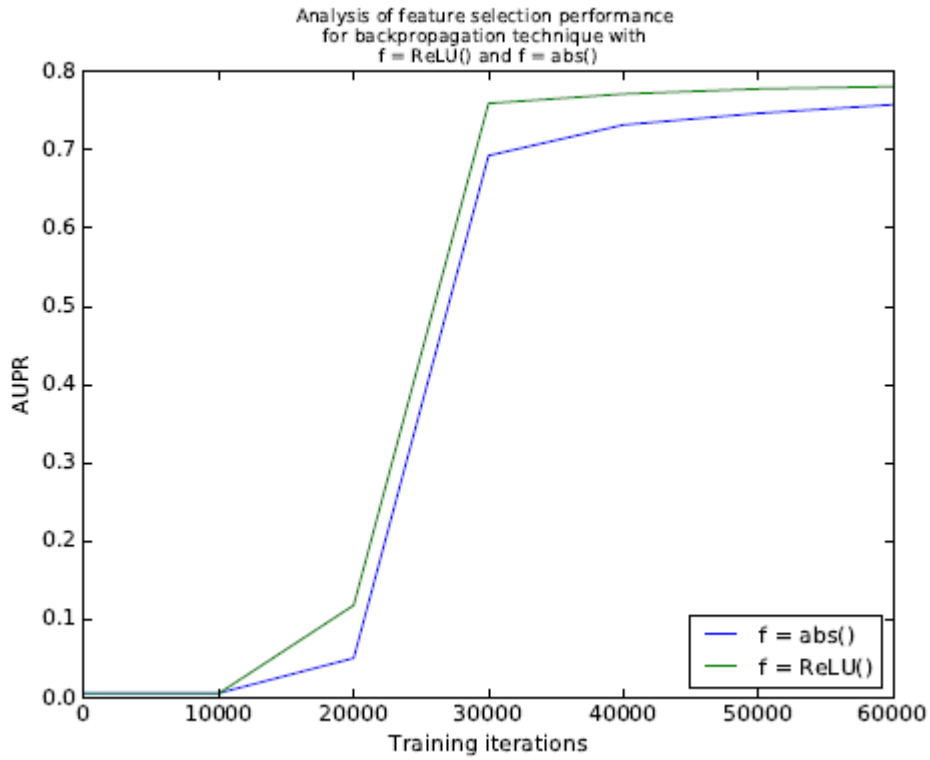

Figure 1: Comparison between feature selection performance

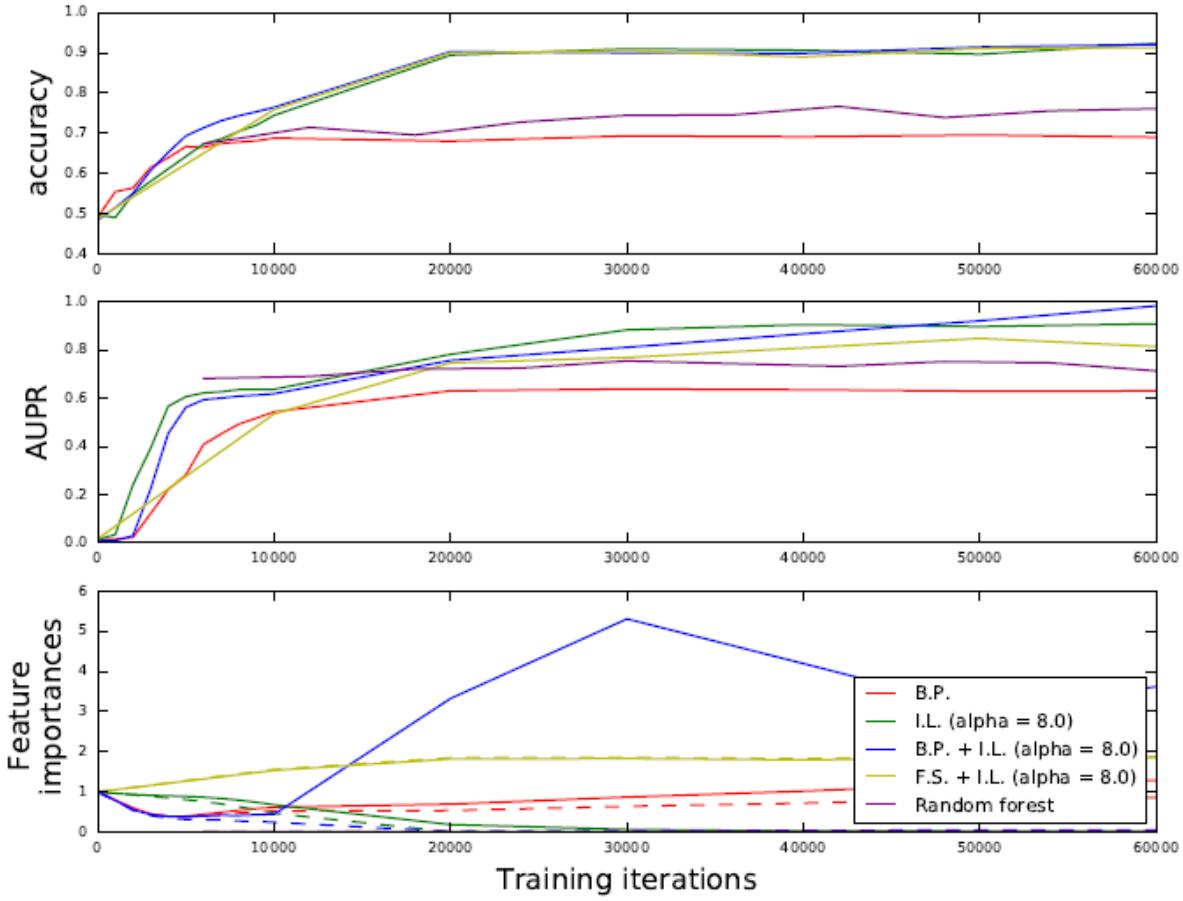

Figure 2: Accuracy, AUPR, and weights evolution of a network for different algorithms

5. Conclusion

Multiple algorithms have been presented with their advantages and drawbacks, thechoice of which to use is thus situation dependant. If the goal is to select features on anun-noised dataset or to use a pre-trained network, then the B.P. algorithm should be used.Otherwise using the B.P + I.L. technique is the way to go most of the time. It must not beforgotten that as shown in Subsection 4.2.2 neural architecture plays a big role. Indeed,results can vary greatly with the number of hidden layers/neurons per layer. However, weshowed that since the higher the accuracy the better the feature selection, this problemcan be addressed by using cross-validation to find a near-optimal architecture. Finally, weshowed that the computation time of the B.P and B.P + I.L. algorithms is of the orderof an epoch, which is not a huge deal compared to the training time. Therefore,only swapping techniques suffer from their computation time.It is also very important to remember the extensions given for each algorithm.First, remember that the regularization can be modified as explained in Section 3.1according to the problem. For example, it is sometimes considered useful to detectredundant features but can also be detrimental. Also as stated, multiple algorithminitialization methods could be imagined and we have given clues on what couldbe changed to the current algorithm to further enhance the results. Asan example, we have given another way to initialize the B.P. algorithm in the case of binary classification with Equation 3.7, which would likely result in enhancedperformances.

References

[1]. Li, Y., yu Chen, C., and Wasserman, W. W. (2015). Deep feature selection: Theory and application to identify enhancers and promoters. JOURNAL OF COMPUTATIONAL BIOLOGY, pages 1-15.

[2]. Marbach, D., Scha_ter, T., Mattiussi, C., and Floreano, D. (2009). Generating realistic in silico gene networks for performance assessment of reverse engineering methods. Journal of Computational Biology, 16(2):229-239.

[3]. Montavon, G., Lapuschkin, S., Binder, A., Samek, W., and M• uller, K.-R. (2017). Explaining nonlinear classi_cation decisions with deep taylor decomposition. Pattern Recognition, 65:211-222.

[4]. R.S.M. Lakshmi Patibandla, B. Tarakeswara Rao, P. Sandhya Krishna, Venkata Rao Maddumala,(2020),"MEDICAL DATA CLUSTERING USING PARTICLE SWARM OPTIMIZATION METHOD", Journal of Critical Reviews, ISSN- 2394-5125, Vol 7, Issue 6, 2020, Page No. 363-367.

[5]. Patibandla R.S.M.L., Veeranjaneyulu N. (2018), "Survey on Clustering Algorithms for Unstructured Data". In: Bhateja V., Coello Coello C., Satapathy S., Pattnaik P. (eds) Intelligent Engineering Informatics. Advances in Intelligent Systems and Computing, vol 695. Springer, Singapore

[6].

[7]. A.Naresh, R S M Lakshmi Patibandla, VidhyaLakshmi, M.MeghanaChowdary.(2020). "Unsupervised Text Classification for Heart Disease Using Machine Learning Methods", Test Engineering and Management, ISSN: 0193-4120 Page No. 11005 – 11016.

[8]. Tarakeswara Rao B., Patibandla R.S.M.L., Murty M.R. (2020) A Comparative Study on Effective Approaches for Unsupervised Statistical Machine Translation. In: Bhateja V., Satapathy S., Satori H. (eds) Embedded Systems and Artificial Intelligence. Advances in Intelligent Systems and Computing, vol 1076. Springer, Singapore.

[9]. RSM Lakshmi Patibandla. (2020), "Regularization of Graphs: Sentiment Classification", Recommender System with Machine Learning and Artificial Intelligence: Practical Tools and Applications in Medical, Agricultural and Other Industries, John Wiley & Sons, pp:373.

[10]. Murugan, R., Paliwal, M., Patibandla, R.S.M.L., Shah, P., Balaga, T.R., Gurrammagari, D.R., Singaravelu, P., (...), Jhaveri, R. Amalgamation of Transfer Learning and Explainable AI for Internet of Medical Things (2024) Recent Advances in Computer Science and Communications, 17 (4), art. no. e191223224674, pp. 40-53. doi: 10.2174/0126662558285074231120063921

[11]. X. Xu, R. S. M. Lakshmi Patibandla, A. Arora, M. Al-Razgan, E. Mahrous Awwad and V. Omollo Nyangaresi, "An Adaptive Hybrid (1D-2D) Convolution-Based ShuffleNetV2 Mechanism for Irrigation Levels Prediction in Agricultural Fields With Smart IoTs," in IEEE Access, vol. 12, pp. 71901-71918, 2024, doi: 10.1109/ACCESS.2024.3384473.

[12]. S. Bhatnagar et al., "Efficient Logistics Solutions for E-Commerce Using Wireless Sensor Networks," in IEEE Transactions on Consumer Electronics, doi: 10.1109/TCE.2024.3375748.

[13]. Krishna, P. S., Reddy, U. J., Patibandla, R. L., & Khadherbhi, S. R. (2020). Identification of lung cancer stages using efficient machine learning framework. Journal of Critical Reviews, 7(6), 385-390.

[14]. Banavathu Mounika, S. R. K., Maddumala, V. R., & Patibandla, R. L. (2020). Data distribution method with text extraction from big data. Journal of Critical Reviews, 7(6), 376-380. Patibandla, R. S. M., & Veeranjaneyulu, N. (2022). A SimRank based ensemble method for resolving challenges of partition clustering methods. Journal of Scientific & Industrial Research, 79(4), 323-327.

[15]. Patibandla, R.S.M.L., Rao, B.T., Rao, D.M., Ramakrishna Murthy, M. (2024). Reshaping the Future of Learning Disabilities in Higher Education with AI. In: Kaluri, R., Mahmud, M., Gadekallu, T.R., Rajput, D.S., Lakshmanna, K. (eds) Applied Assistive Technologies and Informatics for Students with Disabilities. Applied Intelligence and Informatics. Springer, Singapore. https://doi.org/10.1007/978-981-97-0914-4_2

[16]. Gadde, S., Rao, G. S., Veesam, V. S., Yarlagadda, M., & Patibandla, R. S. M. (2023). Secure Data Sharing in Cloud Computing: A Comprehensive Survey of Two-Factor Authentication and Cryptographic Solutions. Ingénierie des Systèmes d'Information, 28(6).

[17]. Patibandla, R. L., Rao, B. T., Murthy, M. R., & Bhuyan, H. K. (2024). Xai-based autoimmune disorders detection using transfer learning. In Machine Learning in Healthcare and Security (pp. 119-129). CRC Press.

[18]. Lakshman Narayana, V., Lakshmi Patibandla, R.S.M., Pavani, V., Radhika, P. (2024). Optimierte naturinspirierte Rechenalgorithmen zur Erkennung von Lungenerkrankungen. In: Raza, K. (eds) Von der Natur inspirierte intelligente Datenverarbeitungstechniken in der Bioinformatik. Springer, Singapore. https://doi.org/10.1007/978-981-99-7808-3_6

[19]. Narayana, V. L., M. Lakshmi Patibandla, R. S., Rao, B. T., & Gopi, A. P. (2022). Use of Machine Learning in Healthcare. Advanced Healthcare Systems: Empowering Physicians With IoT-Enabled Technologies, 275-293. https://doi.org/10.1002/9781119769293.ch13

[20]. Lakshmi Patibandla, R.S.M., Yaswanth, A., Hussani, S.I. (2023). Water-Body Segmentation from Remote Sensing Satellite Images Utilizing Hierarchical and Contour-Based Multi-Scale Features. In: Marriwala, N., Tripathi, C., Jain, S., Kumar, D. (eds) Mobile Radio Communications and 5G Networks. Lecture Notes in Networks and Systems, vol 588. Springer, Singapore. https://doi.org/10.1007/978-981-19-7982-8_21

[21]. Patibandla, R. L., Narayana, V. L., & Gopi, A. P. (2021). Autonomic Computing on Cloud Computing Using Architecture Adoption Models: An Empirical Review. Autonomic Computing in Cloud Resource Management in Industry 4.0, 195-212.

[22]. Lakshman Narayana, V., Rao, G.S., Gopi, A.P., Lakshmi Patibandla, R.S.M. (2022). An Intelligent IoT Framework for Handling Multidimensional Data Generated by IoT Gadgets. In: Al-Turjman, F.,

Nayyar, A. (eds) Machine Learning for Critical Internet of Medical Things. Springer, Cham. https://doi.org/10.1007/978-3-030-80928-7_9

[23]. Patibandla, R. L., Rao, B. T., Krishna, P. S., & Maddumala, V. R. (2020). Medical data clustering using particle swarm optimization method. Journal of Critical Reviews, 7(6), 363-367.

[24]. R. S. M. Lakshmi Patibandla, V. S. Srinivas, S. N. Mohanty and C. Ranjan Pattanaik, "Automatic Machine Learning: An Exploratory Review," 2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Noida, India, 2021, pp. 1-9, doi: 10.1109/ICRITO51393.2021.9596483.

[25]. Narayana, V.L., Gopi, A.P., Patibandla, R.S.M. (2021). An Efficient Methodology for Avoiding Threats in Smart Homes with Low Power Consumption in IoT Environment Using Blockchain Technology. In: Choudhury, T., Khanna, A., Toe, T.T., Khurana, M., Gia Nhu, N. (eds) Blockchain Applications in IoT Ecosystem. EAI/Springer Innovations in Communication and Computing. Springer, Cham. https://doi.org/10.1007/978-3-030-65691-1_16

[26]. Patibandla, R.S.M.L., Narayana, V.L., Mohanty, S.N. (2021). Need of Improving the Emotional Intelligence of Employees in an Organization for Better Outcomes. In: Nandan Mohanty, S. (eds) Decision Making And Problem Solving. Springer, Cham. https://doi.org/10.1007/978-3-030-66869-3_5

[27]. Lakshmi Patibandla R.S.M., Aienala, Lavanya, Alla, Hemasai Sri (2022). Rainfall Extrapolation through Machine Learning, Workshop on Advances in Computational Intelligence, its Concepts and Applications, ACI 2022, Volume 3283, ISSN: 1613-0073 ,pp: 307 – 315,