Scalable Curriculum Learning for Artificial Neural Networks

Mermer, Melike Nur and Amasyali, Mehmet Fatih

Abstract: ***Learning process of people usually starts with easy samples and goes towards hard ones. Using this method for machine learning is called curriculum learning. Samples are given in an order related to their difficulty level, rather than in random order. The aim of this approach is to create models that have better generalization performance. In existing studies, difficulty levels of the samples were determined by prior knowledge and given to the system. However, this is not a scalable approach for every application. Because of that, such studies were usually carried out in very limited application areas. In this study, a new approach is proposed that automatically generates difficulty levels of the samples from data sets. In this way, it is possible to overcome mentioned constraint in the implementations. Thus, curriculum and anti-curriculum learning methods could be applied on many different application areas. In the experiments where artificial neural networks are used as learners, more successful results were obtained with curriculum and anti-curriculum learning compared with the models where samples were given in random order. After various methods have been tried for determining the difficulty ratings of the samples, this study showed that ensemble learning-based approach is more successful.***

Index Terms: ***Curriculum learning, Difficulty level determination, Ensemble learning, K-nearest neighbor, Neural networks.***

1. INTRODUCTION

A planned process is required for people to recognize the world. We get the information which we need during our life in the earlier ages and add new information to our knowledge by using them. Understanding the information that becomes increasingly complex depends on our knowledge about fundamentals. This natural course in the learning process of people is also thought to be beneficial in the machine learning methods. This idea is called curriculum learning. Elman, in his 1993 study [1], proposed the idea of learning simpler parts of a subject at the start and gradually increasing the degree of difficulty. He taught grammar rules to a recursive network in such a way that restricts the complexity at start and expands the architecture, while learning the resources gradually. Results of his work empirically proved that learning is more successfully performed this way. Also, Krueger and Dayan have shown in their work [2] that it is possible to have a significant time gain by separating the whole complex task to sub-components. Similar ideas have arisen in a work in the field of robotics [3]. In the context of these developments, Bengio et al. have added the Curriculum Learning concept to machine learning literature [4].

We trained the learner by considering the difficulty levels of examples in this study. We applied two learning regimes, first one is curriculum (from easy to hard) and the second is anti-curriculum (from hard to easy). Before starting to train the learner we determined difficulty levels of the examples. In their work, Bengio et al. have taken the difficulty level as preliminary information, e. g. the samples which are indicating less diversity in terms of a feature are easy and more diversified examples are difficult. In Karpathy and Van de Panne's work [5], the abilities of a robot have been taught in a sequential manner. The abilities which are combination of several different movements were taught after the abilities that consist only of one single movement. This approach has enabled the robot to use the experience acquired from simple movements on the complex ones.

It may not always be appropriate to decide the degree of difficulty by prior knowledge as in the mentioned studies. Even if a sample is easy for humans, it may not be easy for machines. Because of this fact, Kumar et al. proposed a method [6] that gives to the learner the work of deciding the difficulty degrees. We used two different machine learning methods to determine the difficulty levels in this work. Curriculum and anti-curriculum learning approaches have been applicable in various fields after automatically extracting the difficulty ratings of the samples.

Manuscript received April 7, 2017.

M. N. Mermer is at the Software Engineering Department, Istanbul S. Zaim University, Turkey.
(e-mail: melike.mermer@izu.edu.tr)

M. F. Amasyali is at the Computer Engineering Department, Yildiz Technical University, Istanbul, Turkey.
(e-mail: mfatih@ce.yildiz.edu.tr)

After the samples were divided by difficulty levels, the model has been trained according to the curriculum order (starting from easy samples and going towards hard ones) and anti-curriculum order (starting from hard samples and going towards easy ones). The results are compared with the classical method where we give the samples in random order. We made this comparison on different data sets testing with t-test. The applied approaches and their results are examined.

2. Difficulty Level Determination

Many methods can be used to decide if an example is easy or difficult. In this study, we compared two different methods. First one is k-nearest neighbor and the second is an ensemble algorithm. We used entropy in the output of both methods to determine the difficulty levels.

2.1 K-nearest Neighbor

An example is classified as the most common class of the nearest k training samples in this method [7]. We examined the class distributions of 7-nearest neighbors for difficulty level determination. According to this approach, if the class of the sample is same as the samples around it, it is an easy sample. The sample is considered difficult if it belongs to a different class from the samples around it. For the sample x_p, probability of each class P_i for the label ϑ_i has been calculated using the equation (1).

$$P_i \leftarrow \frac{\sum_{j-1}^{k}\delta(\vartheta_i, f(x_j))}{k} \tag{1}$$

where x_j denotes the j^{th} nearest neighbor of x_p, $f(x_j)$ is the class of x_j, and $\delta(.)$ is the equality function which generates 1 if $f(x_j) = \vartheta_i$, otherwise 0.

2.2 Ensemble Learning

In machine learning methods combining the decisions of multiple classifiers instead of a single classifier ensures higher accuracy rates [8]. Here, estimates of a 10-tree classifier ensemble, which is generated by Breiman's method [9], is used to determine the difficulty degrees of the samples. According to this approach, if classifiers predict the same class for a sample, it is determined as easy. If classifiers predict different class for a sample, it is determined as difficult. Probability for each class P_i for the label ϑ_i is calculated by (2).

$$P_i \leftarrow \frac{\sum_{j=1}^{m}\delta(\vartheta_i, \varphi_j(x_p, L_j))}{m} \tag{2}$$

where m is the number of classifiers, $\varphi_j(x_p, L_j)$ denotes the prediction of the j^{th} classifier works on the learning set L_j, and $\delta(.)$ is the equality function which generates 1 if $\varphi_j(x_p, L_j) = \vartheta_i$ otherwise 0.

2.3 Entropy

Entropy is a measure for homogeneity [7]. In this study, we used entropy for the output of the both methods. In the k-nearest neighbor method, entropy is used for measuring the homogeneity of classes of the neighbors. Low entropy means sample is in the same class with its neighbors. In the ensemble method entropy is used for measuring the homogeneity of classifier predictions. Low entropy means classifiers have agreed about the class of the sample. Thus, for both methods, if a sample has low entropy value (high homogeneity), it is an easy sample; and if it has high entropy (low homogeneity), it is difficult. Entropy of the sample x_p is calculated by (3) where c denotes the number of classes.

$$H(x_p) = -\sum_{i=1}^{c} P_i log_2 P_i \tag{3}$$

We have drawn sample difficulty histograms of some data sets in Fig 1 for both methods. These histograms show the number of samples for each entropy value in the training set. We divided these histograms into 3 equal intervals. The samples

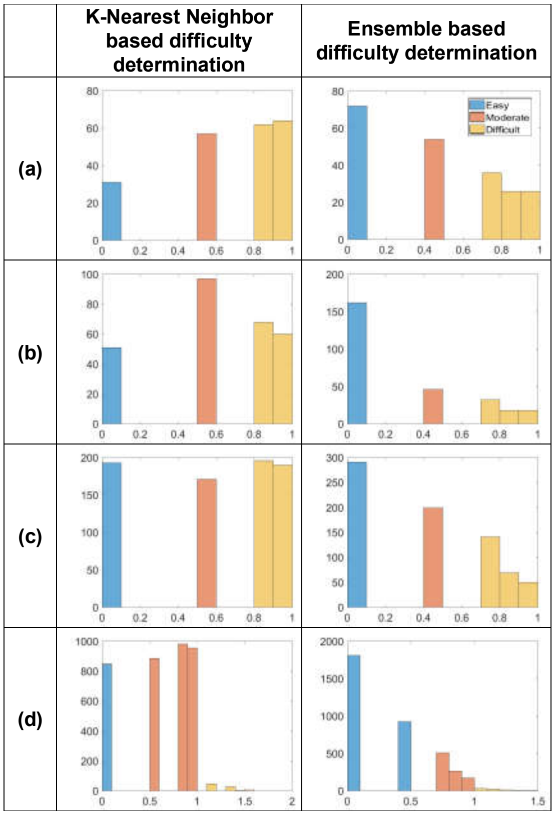

Fig 1. Histograms of (a) breast-cancer, (b) colic, (c) credit-g, (d) waveform data sets for both difficulty determination methods.

that have low entropy are determined as easy, middle samples as moderate, and samples with high entropy as difficult. In Fig 1 the intervals are shown with different colors.

We obtained better leveling in the multiclass data sets by using entropy instead of comparing the output of the difficulty determination methods with the real class of the sample. The samples are leveled relative to each other by specific threshold values for data sets.

3. Preparation of The Training Sets and Training The Model

Curriculum and anti-curriculum learning approaches are experimented on artificial neural networks. We used a feedforward multi-layer perceptron with 3 hidden layers. Numbers of neurons in the hidden layer are determined as 10, 5 and the class count of the data set respectively. The network is trained with gradient descent with momentum. We use incremental method by updating the weights in each sample. One epoch is completed by presenting all samples one at the time.

For the curriculum learning, we designed the training process as three consecutive stages. After the determination of difficulty levels (easy, moderate and difficult) for each sample described as Section 2, samples are presented to the network according to their levels. In Table 1, used samples and their epoch numbers at each stage are given.

Table 1. The stages of curriculum learning

Stage ID	Used difficulty level and epoch number
1	easy (n/3)
2	moderate (2*n/3), easy (n/3)
3	difficult (n), moderate (n/3), easy (n/3)

In Table 1, n is the average epoch number of the classical (random order) method. When all stages are completed, all samples will be presented to the network n times as in the classical method. The previous stage samples are presented again in each stage in order to ensure that the success of the trained network for the old samples is not adversely affected while new samples are being added.

In the case of anti-curriculum learning, the process of curriculum training is performed in inverse order. In this way, the network is trained with decreasing difficulty levels.

4. Experimental Study and Test Results

In order to see the effect of curriculum and anti-curriculum learning approaches in different application areas, experiments are made on various data sets and compared with the classical method.

4.1 Data Sets

The experiments are performed on 36 UCI data sets [10] given in Table 2 with their average convergence epoch. Convergence condition is the raise of the validation set error for 6 epochs. Average of 20 random seeds are calculated.

Table 2. Data sets

ID	Data set	# of Samples	# of Features	# of Classes	Avg. Conv. Epoch
1	labor	57	26	2	21
2	zoo	84	16	4	33
3	lymph	142	37	2	21
4	iris	150	4	3	30
5	hepatitis	155	19	2	21
6	audiology	169	69	5	23
7	autos	202	71	5	17
8	glass	205	9	5	15
9	sonar	208	61	2	21
10	heart-statlog	270	13	2	20
11	breast-cancer	286	38	2	12
12	primary-tumor	302	23	11	12
13	ionosphere	351	33	2	21
14	colic	368	60	2	15
15	vote	435	16	2	24
16	balance-scale	625	4	3	24
17	soybean	675	83	18	14
18	credit-a	690	42	2	12
19	breast-w	699	9	2	15
20	diabetes	768	8	2	21
21	vehicle	846	18	4	29
22	anneal	890	62	4	32
23	vowel	990	11	11	24
24	credit-g	1000	59	2	13
25	col10	2019	7	10	20
26	segment	2310	18	7	24
27	splice	3190	287	3	15
28	kr-vs-kp	3196	39	2	27
29	hypothyroid	3770	31	3	26
30	sick	3772	31	2	33
31	abalone	4153	10	19	14
32	waveform	5000	40	3	13
33	d159	7182	32	2	15
34	ringnorm	7400	20	2	23
35	mushroom	8124	112	2	15
36	letter	20000	16	26	12

4.2 Overview of Difficulty Determination Methods

Prepared training sets are used for curriculum and anti-curriculum learning and these methods are compared with the classical method. Training is restricted with a certain number of epochs in order to ensure that the numbers of presentations of the training samples to be shown in the methods are equal. This number is obtained by stopping the training with a condition that the error in the validation set raises for 6 epochs in the classical method by taking the average of 20 seeds for each data set.

Fig 2 shows the error rates in the methods limited with twofold of the average epoch number for some data sets. The error rates obtained for both k-nearest neighbor and ensemble based curriculum and anti-curriculum learning methods are given. The values in the graphs are average of 20 errors obtained with the methods. It is seen that average error rates of the curriculum and anti-curriculum learning methods are usually lower than classical method. It can also be said that errors of the k-nearest neighbor based curriculum learning approach is decreasing slowly during the training. K-nearest neighbor based curriculum and anti-curriculum learning approaches have higher errors than classical method in some cases such as (c).

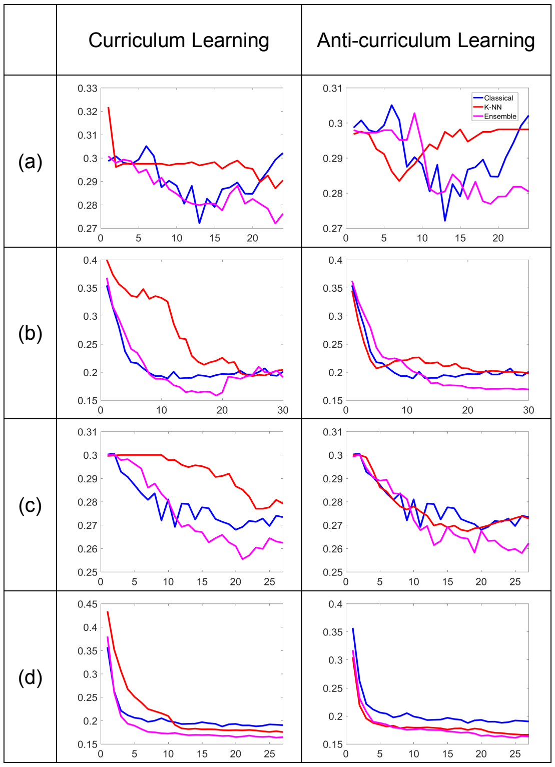

Fig 2. Average test errors at each epoch for proposed methods on (a) breast-cancer, (b) colic, (c) credit-g, (d) waveform data sets.

4.3 Review on Training and Testing Set Errors

We point Fig 3 to understand the relation between training and testing set errors during training. Here we showed training and testing set errors of some data sets for ensemble based difficulty determination during training. In the classical method, while training set error continues to decrease, testing set error is increasing in some cases such as (a). This means the network has been over-trained until it reaches epoch limit. It is seen that training set errors of curriculum and anti-curriculum approaches are usually higher than of classical method. Training set errors of curriculum and anti-curriculum approaches are fixed at near a value and do not go more below this value for determined epoch limit. Presenting different samples at each stage ensures that the network does not get over trained. Thus, testing set errors of curriculum and anti-curriculum learning methods are lower than classical method.

Comparing the 3 methods show that curriculum and anti-curriculum approaches have obtained better error rates at the end of the training on many data sets. Anti-curriculum learning has better results on more data sets and errors of this approach have been decreasing smoother on both training and testing sets.

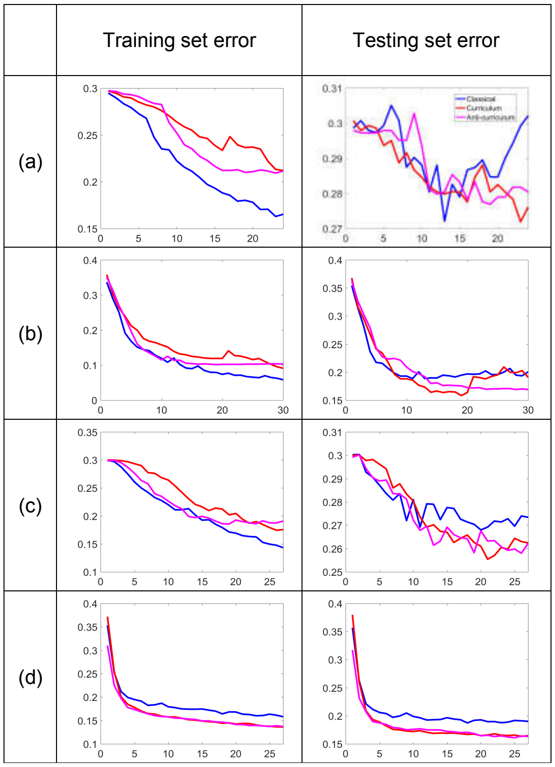

Fig 3. Average training and testing set errors at each epoch for ensemble based difficulty level determination on (a) breast-cancer, (b) colic, (c) credit-g, (d) waveform data sets.

4.4 Comparison Tests and Results

We applied two conditions for each proposed approach. If we assume n is the average epoch number of 20 different seeds which stopped by validation set error increasing 6 times, training limit of the first condition is n and the second is 2n. In this way we designed 6 methods and compared them with t-test on 36 UCI data sets.

Listed 6 algorithms have been compared:

- Classical method, stopped with the condition that validation error is raised for 6 epochs
- Classical method, limited with twofold of the average epoch number (Classical(2n))
- Curriculum learning, limited with the average epoch number (Curriculum(n))
- Curriculum learning, limited with twofold of the average epoch number (Curriculum(2n))
- Anti-curriculum learning, limited with the average epoch number (Anti-curriculum(n))
- Anti-curriculum learning, limited with twofold of the average epoch number (Anti-curriculum(2n))

We obtained 20 different results by 5x4 fold cross validation and different initial conditions of the network. All the methods were started with the same initial conditions for each training.

First, we made comparisons for both difficulty determination methods. In Table 3 we give the results of the comparisons with classical method for ensemble and k-nearest neighbor based curriculum and anti-curriculum approaches limited with twofold of the average epoch number. Win and loss numbers in the cells corresponds to the statistically significant result obtained data set count by t-test. It is seen that the ensemble based technique is more successful. Ensemble based anti-curriculum learning has obtained better results on an important part of the cases. Pay attention that while k-nearest neighbor based difficulty determination method has been lost on some data sets in both curriculum and anti-curriculum approaches ensemble based method either won or tied against classical method.

Table 3. Comparisons with classical method

Difficulty Determination Methods		**# of wins**	**# of ties**	**# of losses**
K-nearest neighbor	**Curriculum**	7	26	3
	Anti-curriculum	11	21	4
Ensemble learning	**Curriculum**	11	25	0
	Anti-curriculum	14	22	0

K-nearest neighbor method decides difficulty by considering locations of the samples. It means if a sample has neighbors from other classes it has been more difficult than a sample whose neighbors from its own class automatically. Another aspect, ensemble method decides difficulty by predictions of the base learners. By this way, a sample is determined as a difficult sample if at least one classifier predicted wrong class for it. Therefore, it can be said that ensemble method produces more reliable difficulty levels.

Secondly, we give the results of the mentioned 6 algorithms for ensemble based difficulty determination method which has been more successful in the previous experiments on 36 data sets in Table 4. Win and loss numbers in the cells are corresponding to the data set counts that we obtained statistically significant error rates by t-test. In the methods, which are limited with twofold of the average epoch number, it is ensured that the samples at each stage have been learned sufficiently and then passed to the next stage. Curriculum and anti-curriculum learning methods have obtained statistically significant error rates against the classical method on more data sets by this way. When we examine the comparison between curriculum learning methods, the method with increased epoch limit has been successful on more data sets. For the classical method, it is seen that there is no general benefit of presenting the samples more.

On a few data sets, the errors obtained by Classical(2n) were higher than the method, which was stopped with the increasing validation set error condition. In the classical method, 2n epoch limit causes over-training in these data sets. For the curriculum and anti-curriculum learning methods limited with the same number of epochs, over training is not the issue.

Here the reason for the success of curriculum and anti-curriculum learning should not be considered as the over-training of the classical method. Because these approaches also achieve significant success against the classical method that is stopped by the validation error risen 6 times. As it is seen in Table 4 ensemble based anti-curriculum learning(2n) has obtained statistically significant errors on 14 data sets both the network stopped by increasing validation set

Table 4. Comparisons of the 6 algorithms

T-test Results[a]		Methods(X)					
		Classical	Classical (2n)	Curriculum (n)	Curriculum (2n)	Anti-curriculum (n)	Anti-curriculum (2n)
Methods(Y)	Classical	-	1/33/2	5/30/1	**10/26/0**	4/32/0	**14/22/0**
	Classical (2n)	2/33/1	-	6/29/1	**11/25/0**	6/30/0	**14/22/0**
	Curriculum (n)	1/30/5	1/29/6	-	8/28/0	4/31/1	11/25/0
	Curriculum (2n)	0/26/10	0/25/11	0/28/8	-	0/32/4	2/33/1
	Anti-curriculum (n)	0/32/4	0/30/6	1/31/4	4/32/0	-	4/32/0
	Anti-curriculum (2n)	0/22/14	0/22/14	0/25/11	1/33/2	0/32/4	-

[a] X(win)/tie/Y(win)

error criteria and the network stopped at twofold of the calculated average epoch number. Ensemble based curriculum learning(2n) has also statistically significant performance on many data sets.

5. Conclusions

In this paper, we proposed two different methods that automatically generate difficulty levels of the samples from data sets to overcome the problem of difficulty level determination that causes only a limited scope for curriculum learning. We have seen that using predictions of the classifiers of an ensemble is better than using classes of the nearest neighbors for deciding the difficulty degrees.

Curriculum and anti-curriculum approaches have been practicable for many different fields after resolving the difficulty level determination issue. Thus, we made our experiments on various application areas. We have compared our curriculum and anti-curriculum learning approaches with the classical method. Results of the comparisons have shown that presenting the input samples in an order related with their difficulty level instead of random order is an efficient procedure on many of the application areas. We have obtained better results than classical method on 11 data sets with curriculum learning. We also obtained better results with anti-curriculum learning on 14 data sets.

It is considered that curriculum learning approach obtains a better local minimum at the end of the training. There are some studies [11, 12] about how could curriculum learning approach performs better and what is in the background of it. On the other hand, it is plausible to have better accuracies when we give increasingly difficult samples to the learner as in the curriculum learning. However anti-curriculum learning approach is also fine and even more successful according to our study. It is an interesting subject that should be investigated why anti-curriculum learning obtains better results.

References

[1] Elman, J. L., "Learning and development in neural networks: The importance of starting small", Cognition, Vol. 48, 1993, pp. 781-799.

[2] Krueger, K. A., Dayan, P., "Flexible shaping: How learning in small steps helps", Cognition, Vol. 110, 2009, pp. 380-394.

[3] Sanger, T. D., "Neural network learning control of robot manipulators using gradually increasing task difficulty", IEEE Transactions on Robotics and Automation, 1994, pp. 323-333.

[4] Bengio, Y., Louradour, J., Collobert, R. and Weston, J., "Curriculum Learning", ICML '09 Proceedings of the 26th Annual International Conference on Machine Learning, 2009, pp. 41-48.

[5] Karpathy, A., Van de Panne, M., "Curriculum learning for motor skills", Canadian Conference on Artificial Intelligence, 2012, pp. 325-330.

[6] Kumar, M. P., Packer, B., Koller, D., "Self-paced learning for latent variable models", In Advances in Neural Information Processing Systems, 2010, pp. 1189-1197.

[7] Mitchell, T., Machine Learning, *McGraw-Hill Science*, 1997.

[8] Opitz, D., Maclin, R., "Popular ensemble methods: An empirical study". Journal of Artificial Intelligence Research, Vol. 11, 1999, pp. 169–198.

[9] Breiman, L., "Bagging predictors", Machine Learning, Vol. 24, No. 2, 1996, pp. 123-140.

[10] UCI Machine Learning Repository, Retrieved from https://archive.ics.uci.edu/ml/datasets.html.

[11] Gong T., Zhao Q., Meng D., Xu Z., "Why curriculum learning & self-paced learning work in big/noisy data: a theoretical perspective", American Institute of Mathematical Sciences Big Data and Information Analytics, Vol. 1, No. 1, 2016, pp. 111-127.

[12] Meng D., Zhao Q., Jiang L., "What objective does self-paced learning indeed optimize?", arXiv preprint arXiv: 1511.06049, 2015.