A New Method in Feature Selection based on Deep Reinforcement Learning in Domain Adaptation

Hala A. Naman[1]*, Zinah Jaffar Mohammed Ameen[2]

[1]College of Engineering, Wasit University, Wasit, Iraq
[2]Computer Engineering, University of Technology, Baghdad, Iraq

Abstract

In data mining and machine learning methods, it is traditionally assumed that training data, test data, and the data that will be processed in the future, should have the same feature space distribution. This is a condition that will not happen in the real world. In order to overcome this challenge, domain adaptation-based methods are used. One of the existing challenges in domain adaptation-based methods is to select the most efficient features so that they can also show the most efficiency in the destination database. In this paper, a new feature selection method based on deep reinforcement learning is proposed. In the proposed method, in order to select the best and most appropriate features, the essential policies in deep reinforcement learning are defined, and then the selection features are applied for training random forest, k-nearest neighborhood and support vector machine classifiers. The trained classifiers with the considered features are evaluated on the target database. The results are evaluated with the criteria of accuracy, sensitivity, positive and negative predictive rates in the classifiers. The achieved results show the superiority of the proposed method of feature selection when used in domain adaptation. By implementing the RF classifier on the VisDA-2018 database and the Syn2Real database, the classification accuracy in the feature selection of the proposed deep learning reinforcement has increased compared to the two-feature selection of Laplace monitoring and feature selection states. The classification sensitivity with the help of SVM classifier on the Syn2Real databases had the highest values in the feature selection state of the proposed deep learning reinforcement. The obtained number 100 is a positive predictive rate in the Syn2Real database with the help of SVM classifier and in the case of selecting the proposed feature, it indicates its superiority. The negative predictive rate in the Syn2Real database in the state of feature selection of the proposed deep reinforcement learning was 100%, which showed its superiority in comparison with 90.1% in the state of selecting the Laplace monitoring feature. Gmean in KNN classifier on the Syn2Real database has improved in the feature selection state of the proposed deep learning reinforcement in comparison to without feature selection state.

Keywords: Domain adaptation, Feature selection, Pattern Recognition, Classification, Deep reinforcement learning.

<div dir="rtl">

تحسين تصنيف معايير التقييم بالاعتماد على طرق تكييف المجال

هالة عبد العظيم نعمان * , زينة جعفر محمد أمين[2]

[1]كلية الهندسة, جامعة واسط , اواسط, العراق

</div>

_____

*Email: haltaee@uowasit.edu.iq

²هندسة الحاسوب, الجامعة التكنولوجية , بغداد, العراق

الخلاصة

في أساليب التنقيب عن البيانات والتعلم الآلي ، يُفترض تقليديًا أن تكون بيانات التدريب ، بيانات الاختبار و بالمثل البيانات التي ستتم معالجتها في المستقبل يكون لها نفس توزيع مدى الميزات. هذه الحالة لن تحدث في العالم الحقيقي. من أجل التغلب على هذا التحدي ، يتم استخدام أساليب تعتمد على مجال التكيف. يتمثل أحد التحديات الحالية في الأساليب القائمة على تكييف المجال في اختيار الميزات الأكثر كفاءة بحيث يمكنها أيضًا إظهار أكبر قدر من الكفاءة في قاعدة البيانات. سيتم اقتراح طريقة اختيار ميزة جديدة تعتمد على التعلم المعزز العميق. في الطريقة المقترحة ، من أجل تحديد أفضل الميزات وأكثرها ملاءمة ، سيتم تحديد السياسات الأساسية في التعلم المعزز العميق ، ثم سيتم تطبيق ميزات الاختيار لتدريب المصنفات العشوائية للغابات ، و k–الأقرب ، ودعم مصنفات آلة المتجهات. سيتم تقييم المصنفات المدربة مع الميزات المدروسة في قاعدة البيانات الهدف. سيتم تقييم النتائج بمعايير الدقة والحساسية والمعدلات التنبؤية الإيجابية والسلبية في المصنفات. تظهر النتائج المحققة تفوق الطريقة المقترحة لاختيار الميزة من أجل استخدامها في تكييف المجال

1.   Introduction

Machine learning is one of the most widely used branches of artificial intelligence, which creates algorithms based on which systems can learn. These algorithms allow the system to use and learn data to improve various functions. In most machine learning algorithms, it is assumed that the training domain and test domain data follow the same distribution. However, in inter-domain and real-world problems, this condition is not met, and the classification model created in the training domain will have low accuracy in predicting the labels of the test domain samples. For example, in segmentation and classification cases, one domain has sufficient data, while in other domains perhaps there is no any data at all, or there may not be enough data, or the data attribute space is completely different. In this case, if the methods based on domain adaptation or knowledge transfer perform properly, learning efficiency will be increased so well. Since the labelling of training data is very expensive and costly, the use of domain adaptation-based methods will be very helpful [1].

However, data mining and machine learning has been used in many fields and has brought a great success, this weakness, i.e., the similar feature space in training data and test data and real-world validation data in real world, is considered a major challenge. When the feature distribution is changed in the feature space, statistical models are required to be able to estimate the new feature space and of course, it is very expensive. In order to overcome this challenge, the methods based on domain adaptation have been suggested. The need for methods based on time domain adaptation shows that training data is insufficient. In this case, the labelled data is so few, or this data is not identical to the new data that is recorded [2].

Recently, the available data, both in number and size, have increased significantly in many machine learning applications. Based on knowledge acquisition, studying the manner of using this large-scale data is very important and necessary. The large amounts of data with huge dimensions have posed a significant challenge to machine learning methods. Because of the existence of noisy, irrelevant and additional data, learning algorithms slow down significantly and reduce the efficiency of learning methods, which also leads to difficulty in the model interpretation. This challenge is also evident in the field of adaptation learning. For this purpose, by choosing the proper features, this challenge can be overcome. Feature selection can select a small subset of related features from the main features by removing irrelevant, noisy and extra features [3].

The exact definition of a feature selection depends on the application field. But the most widely used definition is that feature selection is conducted in such a way that the selection of a subset of features with the best result of the classification performance [4]. The logic for the

mentioned definition is that additional or irrelevant features often have similar noise in the data, which causes the classifier to be mistaken and the classification performance to be downgraded. Omitting such features causes the consequential features having the same or higher classification function than the total features. As a direct result, fewer features are required for storage, and hence, the classification operation is accelerated. Moreover, reducing the number of features helps the expert human to focus on a subset of the relevant features, this subject result in having a better view of the process described by the data. [5].

Generally,, feature selection helps to better understand data, reduce computational requirements, reduction of the destructive effects of dimensions, and improvement of predicted efficiency. Feature selection focuses on selecting a subset of input variables that can effectively describe input data, reduce noise effects and irrelevant variables, while providing the predicted results so well [6]. In [7] used and suggested a sentiment analysis system based on the Bayesian Rough Decision Tree (BRDT) algorithm for machine learning to choose proper features.

Reinforcement learning is one of the machine learning trends that is inspired by behavioural psychology. This method focuses on the behaviours that the machine must perform to maximize its reward. This issue is checked in this domain because of this method scope in different domains like theory of control, theory of game, theory of information, research of operations, genetic algorithm, optimization based on simulation, swarm intelligence, multi-agent system, statistics. In operations research domain and literature of control, the domain in which method of reinforcement learning is studied is known as the accurate dynamic programming [8], while enhancer Deep Belief Network (DBN) as a one of the deep neural networks was being used in [9].

The main purpose of this research is to present a new method in feature selection based on deep reinforcement learning in domain adaptation. In the following sections of this paper: in section 2, the concepts of domain adaptation and feature selection in this domain will be explained. In Section 3, the concepts of deep reinforcement learning will be described. In section 4, the proposed method will be explained. In section 5, the proposed method will be simulated on several databases, and eventually in section 6, the conclusion of the whole article is presented.

2.   Feature Selection in The Domain Adaptation

There are many sets of domain adaptations in the real world of learning. For example, when humans recognize the apple fruit, they also identify the pear fruit with this identification. Similarly, driving a car and learning it can help you drive other cars. The main motivation of the research in adapting the field started with the argument that the knowledge that people learn in a field can solve other problems in similar fields faster and easier. But the main motivation for using domain adaptation was in the field of VIPS95 machine learning. With the topic of " Learning to Learn " in which machine learning methods can be used for later applications. Domain adaptation has been proposed with the names of knowledge adaptation, Life-Long Learning, and Meta learning [10]. Even the discussion of a Multitask Learning framework is mentioned that is similar to domain adaptation. The main task of adapting the field is the ability of a system to detect and apply knowledge and proficiency from previous works to new works. In the field of mission and definition of domain adaptation, its purpose is to learn and extract knowledge from one or more tasks and source task and apply it to a target task and the task and the role of tasks and work in the source and target are not very symmetrical.

Figure 1 shows the difference between the traditional learning process and domain adaptation. Traditional domain adaptation methods learn each task and work in each set and use it in the same domain. This is in the case that domain adaptation methods use the learning

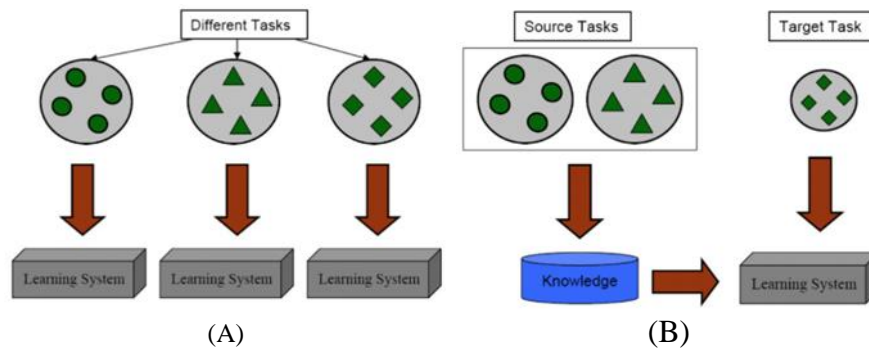of previous tasks in target learning. This is in the condition that they have a few training data [11].

(A)   (B)

Figure 1-Difference between domain and traditional adaptation, a) traditional learning model, b) domain adaptation model [12]

A. Domain Adaptation

To introduce domain adaptation, it is necessary to define completely and clearly the two concepts of task and domain D. Domain D consist of two sections:

- **First section:** Feature Space x
- **Second section:** Marginal Probability distribution of the margin domain adaptation $P(X)$ where $X = \{x_1, \dots x_n\}$

In an intended domain $D = \{X, P(X)\}$, a task consists of two sections:

- **First section:** Y label space
- **Second section:** Objective predictive function is denoted by *f(.)*. A task is represented by *{T={Y,f(.)}* , of course, it is not observed, but it is learned from training data [12], [13].

T contains the pairs {x_i,y_i } that $y_i \in Y$,$x_i \in X$ is the function *f(.)* , which is used for predicting the corresponding labels. For example, *f(x)* for the new case x is written from a statistical point of view *f(.)* in the form of a conditional probability function P(y|x). Since there is an adaptation between the source domain and the source domain, D_S will be the source domain that will be used to learn in the target domain D_T. In this case, the Eq.1 and Eq.2 relationships are established [15].

$$D_S = \{(x_{s_1}, y_{s_1}), \dots, (x_{s_n}, y_{s_n})\} \tag{1}$$
$$D_T = \{(x_{Ts_1}, y_{T_1}), \dots, (x_{T_n}, y_{T_n})\} \tag{2}$$

By having the learning domain D_T and the learning task T_s, a target domain D_T and the target task T_T, the domain adaptation is performed to improve learning in the objective function $f_T(.)$ in D_T, which uses D_s and T_s. It should be noted that $D_s \neq D_T$ or $T_s \neq T_T$ in this definition are $D = \{X, P(X)\}$ resulting in $P_s(X) = P_T(X)$.

For example, they are completely different in the classification and segmentation of features in the training set and the purpose and related features. Furthermore, the pair $T = \{y, P(y|x)\}$, leads to the ambiguous condition $T_s \neq T_T$ that $y_s \neq y_T$ and therefore, $P(y_s|x_s) \neq P(y_T|x_T)$. When the condition of inequality is removed from these definitions, then the intended problem will be a traditional learning problem [15].

B.The Classification of Domain Adaptation

In the domain adaptation, three issues are considered: a) what should be adapted, b) how it should be adapted, c) when it should be adapted.

The problem of "what should be adapted" is looking to determine which part of the knowledge can be adapted to the domain or task. Some knowledge is specific or it is used for a specific domain. But some knowledge can be shared between different areas, and this

sharing can also increase efficiency. Therefore, to identify the reaction that must be adapted, it is essential to design the required training algorithm. It means that, the "problem of the manner of adaptation" is addressed. The question of how the training algorithm adapts must be properly addressed. At the end, in the "adaptation time", the position of adaptation and expertise is considered. In fact, in this case, there is an interest in the topic of what knowledge should be adapted and what knowledge should not be adapted [16].

C.   Selection of Features in Domain Adaptation

Features display the characteristics of objects, and the key solution to identifying objects and classifying them is to efficiently select a combination of features. Such characteristics are given to a classifier and the performance of classification is based on such features' efficiency. Extracting the appropriate features is hard because of a lot of agents like noise. In systems of detection and recognition, choosing feature aim is finding the features subset which leads to the best performance of detection and recognition that require the least effort of computation [17]. Feature Selection is important for the systems of detection and recognition for following reasons:

• Sometimes, many features are accessible for the system of detection/recognition. But such features are dependent or interdependent to each other. The bad feature can decrease the performance of system intensively. Utilizing more features can raise the system complexity; however, it might not lead to the high accuracy of detection/ recognition. So, this is essential for choosing the good features subset [18].

• Features are chosen by the algorithm of learning in the step of training. Chosen features are utilized as the model for describing data of training. Choosing a lot of features means that a complicated model has been utilized for approximating data of training. Based on Minimum Description Length Principle (MDLP), the simple model is better than the complicated model. Data of training might be corrupted by a lot of noises also the complicated model is sensitive to noise in data of training and it has a substandard performance on data of test [19].

• Utilizing the fewer features can decrease the costs of computation which is essential for the applications of real-time. Decreasing the features might develop accuracy of classification [20].

Ideally, the goal is to use features that have better separation power.

3.   Reinforcement Learning

In a reinforcement learning problem, there is a factor that interacts with the environment through trial and error and learns to choose the optimal action for achieving the goal. In this type of learning, there is no external observer, and the agent alone interacts with the environment, learns, and gains experience and receives some rewards. In reinforcement learning, agents are equipped with sensors that can receive the definite characteristics of the environment. These features constitute the learning agent space [20].

Then, at each time interval, the agent affects the environment by performing an operation. Hence, the agent receives different inputs in the next time period based on the previous action. In addition to the new input, the learner receives a reinforcement signal at any time indicating the desirability of the action, which is called a reward This value can be negative or positive, depending on whether the action is suitable or not. In general, all reinforcement learning factors have a transparent purpose. They can understand their environment; choose actions to influence their environment. They interact with their environment, take action and receive rewards. However, they may not fully recognize their surrounding environment [21].

Since the agent interacts with his environment, so its actions are effective in the environment and next situations. So, the agent must periodically monitor its environment and react properly. Reinforcement learning differs from learning with observer in two aspects [21].

Learning examples are not presented in pairs (input / output). But after the agent does an action, it receives a reward and goes to the next stage. The agent has no idea about what the best action is in each state. Over time, the agent collects sufficient experience of states, possible actions, adaptation to new state and reward, and learns optimal performance. Another difference is that the system must have high online efficiency since the system is often evaluated simultaneously with the learning operation. For main features of reinforcement learning, we can refer to the following: The learner is not told what to do. The search is based on trial and error. The learner tries to learn the actions that produce the most rewards. Rewards are of the delay type: hence short-term achievements are sacrificed for longer-term benefits. Fitness must be created between exploring new cases and using previous knowledge. Explore or exploit consider the problem as a purposeful factor related to an unspecified environment [22].

The general algorithm in reinforcement learning is as follows: 1. Observe the current state,2. Deciding to take action, 3. Take action, 4. Receive a reinforcement signal, 5. observe the new state, 6. Learning from experiences, 7. Repetition [23].

4. The Proposed Method in The Feature Selection

The problem of feature selection with the help of deep reinforcement learning algorithm is considered as a sequential decision problem. At each stride, an agent selects a feature to perform the relevant classification or to predict the classification result. In Q-Learning, each agent searches around and its surrounding environment and it examines rewards and situations with its performed activity [24].

In this paper, a new feature selection method based on deep reinforcement learning is proposed. In the proposed method, in order to select the best and most appropriate features, the essential policies in deep reinforcement learning are defined, and then the selection features are applied for training random forest, k-nearest neighbourhood and support vector machine classifiers.

In this research, the policy improvement method was used to select the feature in the deep reinforcement learning algorithm. The proposed improvement method is looking for value estimation of the action- state pair with a function Q. This estimate is done using a neural network. The function Q^π *(s,a)* represents the reward for discounted start in the observed state *s*, which occurs when applying action a and by presenting the policy π after it, which has a recurrent form. As illustrated in Eq.3 [15].

$$Q^\pi(s,a) = r\,(s,a) + E\left[\gamma\,Q^\pi\big(S, \pi\,(S)\big)\right] \qquad (3)$$
$$s \sim t\,(s,a)$$

Note that *r (s, a)* is the expected value in all rewards. The value of the action- state pair in the terminal state is also zero, i.e. *Q (τ ,0)=0.*

The value of the factor γ, which is also a discrete value, determines the importance of future rewards. It is extremely useful in non- episode environments or when the approximation function is used. Because the features considered in this research are the type of discrete features, the intended data environment ensures the terminally of the algorithm. The numerical value of γ is helpful during the training process and is a determining parameter.

In practice, the optimal function Q^* is the desired one, in such a way as illustrated in Eq.4 [15].

$$\pi^*(s) = \ argmax_a Q^*\,(s,a) \qquad (4)$$

in this relation $Q^*(s_1 a)$ can be computed in Eq.5 [15].

$$Q^\pi(s,a) = r\,(s,a) + E\left[\gamma\,Q^\pi\big(S, \pi\,(S)\big)\right] \qquad (5)$$
$$s \sim t\,(s,a)$$

In limited feature spaces and small dimensions as well as low state space, the Q function is easily determined by the dynamic state. But if the exact space has large dimensions or the desired space is continuous, it is almost impossible to determine Q. To overcome this

challenge, the neural network has been used, which results in Deep– Q – Learning [14]. Inspired by neural networks and dynamic programming, a neural network θ can estimate the function Q^θ, which is done by minimizing the Mean square error (MSE) between the two sides of equation (5). For transmission of (s, a, r, s'), which is performed experimentally with the desired agent with a greedy policy, i.e., the Eq.6 [15].

$$\pi^\theta(S) = argmax_a Q^\theta(s,a) \tag{6}$$

In this transmission, s and a represent the performed action and the current state, r represents the obtained reward, which is expected to converge to the value of s (r, a) and also s is the next state, i.e. t(s,a) s˜ in order to explain it with formula, the parameter θ will repeatedly minimize the L_θ loss function can be computed from Eq.7 [15].

$$L_\theta(B) = \frac{1}{|B|}\sum_{(s,a,r,s)\epsilon B}(q(r,s) - Q^\theta(s,a))^2 \tag{7}$$

In relation (7), q is the objective estimation of the function Q, which is constant. For the values of w, r and t: it is constant. In fact, the parameters r, w and t are constant in the optimization step, $q(r,s)$ can be computed by using Eq.8 [15].

$$q(r,s) = r + y \max Q^\theta(s,a) \tag{8}$$

when the error increases, the expected function Q^θ will be converged to Q^*.

In this algorithm, each part with a random sampled value (x, y) of the corresponding training data, which practically defines the starting state, $\tilde{S}_0$. At any given time, a state is $\tilde{S} = (x_1 y_1 \tilde{F})$ where each observed state is produced by $S = \{(x^i f_i \parallel \forall f_i \in \widetilde{F}\}$. The neural network accepts only numerical input values. Therefore, the observed state *s* will be written in Tuple $.(\bar{x}, m)$. The vector $\bar{x} \in R^n$ is the masked vector of the original vector *x*. $\bar{x}_0$ contains the values of *x* that are accepted and defined as illustrated in Eq.9 [15].

$$\bar{x}^{(i)} = \begin{cases} x^i & if & f_i \epsilon \bar{F} \\ 0 & Otherwise \end{cases} \tag{9}$$

The mask $m \in \{0,1\}^n$ is a vector that indicates that a desired feature is accepted and contains the value 1 obtained in the feature position or the value zero (Eq.10 [15]).

$$m^{(i)} = \begin{cases} 1 & if & f_i \in F \\ 0 & otherwise \end{cases} \tag{10}$$

The combination of $m, \bar{x}$ ensures that the network can distinguish between a feature that is not displayed or observed and has a value of zero. To find the optimal Q* function and the correct approximation, that greedy policy and neural network approach, i.e., Deep-Q-Learning, was used. The architecture used in the neural network is in [25] and its parameters are the same for all actions. The target network will consider the observed value as well as the operation performed and the Q value of the output for all states and activities. The target network consists of an input layer, each hidden layer and a fully connected output layer (Figure 2). The activity function of each layer, sigmoid and output layer, is nonlinear.
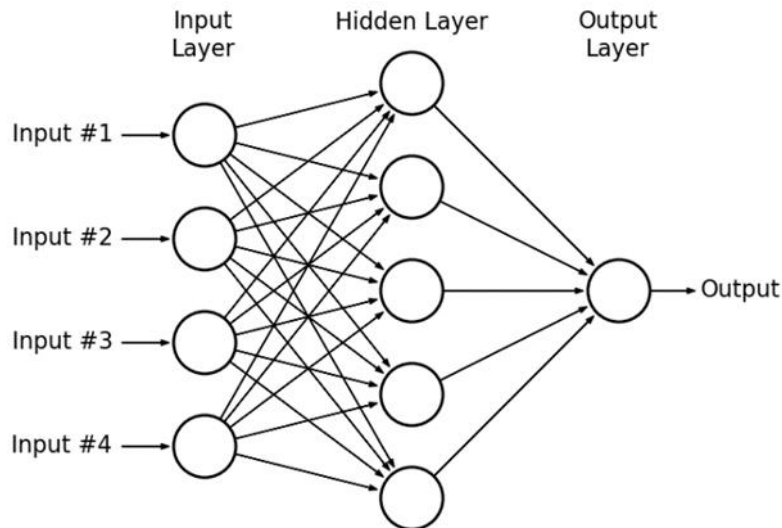
Figure 2-MLP neural network with three input-output layers and one hidden layer

Features are completely normal. Feature programming is done with the help of mean and variance of features. Achieving better and more accurate efficiency of the desired method in separating emails has been performed one hundred times and the result is considered as the final average. The pseudo-code of the proposed method is shown in algorithm 1. In each iteration, one step is simulated in the whole environment and a total of one hundred transitions (S, a, r, s) are generated. The values obtained (S, a, r, s) will be stored in a circular texture M and are randomly sampled in each B batch.

The value of the parameter Y = 1 is considered the desired environment of each Episodic. To optimize the loss function $L_\theta(B)$ in Equation (10) as well as Equation (11), the value of the Momentum parameter s, a is considered and Gradient normalization is considered greater than 1. The training rate varies exponentially and the exact values of the parameters vary based on the number of features. In this research, it depends on the type of email database as well as the number of features. After each optimization, the network weights are updated. This update is done with the help of Equation (11).

$$\phi := (1 - P)\phi + P\phi \tag{11}$$

Randomly initialize parameters θ and ϕ
Initialize environments $\mathcal{E}$ with $(x, y, \emptyset) \in (\mathcal{X}, \mathcal{Y}, \wp(\mathcal{F}))$
Initialize replay buffer $\mathcal{M}$ with a random agent
while not converged **do**
    for all $e \in \mathcal{E}$ **do**
        Simulate one step with ϵ-greedy policy π_θ:
            $a = \pi_\theta(s); \quad s', r = \text{STEP}(e, a)$
        Add transition (s, a, r, s') into circular buffer $\mathcal{M}$
    end for
    Sample a random batch $\mathcal{B}$ from $\mathcal{M}$
    for all $(s_i, a_i, r_i, s_i') \in \mathcal{B}$ **do**
        Compute target q_i:

$$q_i = \begin{cases} r_i & \text{if } s_i' = \mathcal{T} \\ r_i + \gamma Q^\phi(s_i', \underset{a'}{\text{argmax}}\, Q^\theta(s_i', a')) & \text{otherwise} \end{cases}$$

        Clip q_i into $[-1, 0]$
    end for
    Perform one step of gradient descent on ℓ_θ w.r.t. θ:

$$\ell_\theta(\mathcal{B}) = \sum_{i=1}^{|\mathcal{B}|} (q_i - Q^\theta(s_i, a_i))^2$$

    Update parameters $\phi := (1 - \rho)\phi + \rho\theta$
end while

Figure 3-pseudo-code of the proposed method

5.   The Evaluation of Proposed Method

The proposed feature selection method based on deep reinforcement learning for improving the performance of domain adaptation-based approaches can be used with real-world data. The proposed method of feature selection was simulated on two databases, Syn2Real and VisDA-2018, at University of California Irvine (UCI). It should be noted that 70% of the total data was considered as train data and 30% of the total data was considered as test data. The final step in a classification system is its evaluation. In this paper, the criteria of accuracy [10], sensitivity [1], geometric mean [26], negative predictive value and positive predictive value [8] criteria were applied.

A.   Datasets

VisDA2017 [27] dataset focuses on a special domain adaptation setting (simulation to real). The source domain consists of images generated by game engines and target domain consists of real-world images. There are 12 classes in this dataset.

The Syn2Real [28] dataset is constructed to perform object classification in real images by learning from synthetic images. The source images were generated by rendering 3D models of 12 common classes and 33 background classes from different angles and under different lighting conditions. It contains 152,397 synthetic images.

B.   Classifiers

The experiment used three machine learning categories: support vector machine (SVM), nearest neighbour (KNN), and random forest (RF). These methods are suitable for classification tasks. The SVM algorithm analyses the data for regression analysis and classification. Draws this data in a high-dimensional functional space, so that the data set can be categorized even if the data are not separated linearly [29]. KNN is a machine learning algorithm that uses data and identifies new datasets based on similarities. This is done to evaluate the nearest neighbours based on the minimum distance from the test samples to the training data set [30]. Random forest is a method that generates several decision trees at random in a "forest". It creates a large number of decision trees and produces from a class that is a classification or regression of individual trees [31].

1)   The Accuracy Evaluation

The results of classification accuracy evaluation with the help of three classifiers RF, KNN and SVM on the two databases used in the paper in three states of selecting the Laplace monitoring feature, feature selection of the proposed deep learning reinforcement and without feature selection in domain adaptation are shown in Table 2.

Table 2-The evaluation results of accuracy classification criterion

The used database	VisDA-2018			Syn2Real		
The used classifier	RF	KNN	SVM	RF	KNN	SVM
Feature selection of the proposed deep reinforcement learning	87.25	82.11	90.90	78.26	65.21	100
Feature selection of Laplace monitoring	83.65	76.68	89.47	71.23	69.56	83.11
Without feature selection	73.85	74.21	84.21	73.91	70	86.70

By implementing the RF classifier on the VisDA-2018 database and the Syn2Real database, the classification accuracy in the feature selection of the proposed deep learning reinforcement has increased compared to the two-feature selection of Laplace monitoring and feature selection states. The classification accuracy of this classifier on the VisDA-2018 database has improved in the feature selection of the proposed deep learning reinforcement state compared to without feature selection state. Also, for other classifiers in the VisDA-2018 database, the results of the accuracy of the proposed method are better than the other two states. The obtained number 65.21 is accuracy in the Syn2Real database with the help of KNN classifier and the obtained number 82.11 in the case of selecting the Feature selection of the proposed deep reinforcement learning. There are also acceptable results on the VisDA-2018 and Syn2Real databases in the feature selection state of the proposed deep learning reinforcement with the help of SVM classifier.

2) The Evaluation of Classification Sensitivity

Classification sensitivity was evaluated with the help of three classifiers, RF, KNN and SVM, on the two databases used in the paper in three states: selecting the Laplace monitoring feature, selecting the proposed deep reinforcement learning feature, and without feature selection. Table 3 demonstrates the obtained results for sensitivity.

Table 3-The evaluation results of classification sensitivity criterion

The used database	VisDA-2018			Syn2Real		
The used classifier	RF	KNN	SVM	RF	KNN	SVM
Feature selection of the proposed deep reinforcement learning	86.54	31.11	83.25	55.42	59.19	100
Feature selection of Laplace monitoring	73.68	28.57	87.50	53.28	65.60	91.2
Without feature selection	66.66	9.42	80	56.41	54.83	68.88

The classification sensitivity with the help of SVM classifier on the Syn2Real databases had the highest values in in the feature selection state of the proposed deep learning reinforcement. This value has reached 86.54 on the VisDA-2018 database with the help of RF classifier.

3) The evaluation of positive predictive rate

The positive predictive rate of the classification was evaluated with the help of three classifiers RF, KNN and SVM on the two databases used in the article in three states of selecting the Laplace monitoring feature, in the feature selection of the proposed deep learning reinforcement and without feature selection. Table 4 demonstrates the positive predictive rate results.

Table 4-The evaluation results of the positive predictive rate criterion

The used database	VisDA-2018			Syn2Real		
The used classifier	RF	KNN	SVM	RF	KNN	SVM
Feature selection of the proposed deep reinforcement learning	89.36	98.2	90.21	68.53	82.43	100
Feature selection of Laplace monitoring	87.50	100	87.50	63.04	59.72	92.1
Without feature selection	75	100	83.30	73.33	58.66	61.38

The obtained number 100 is a positive predictive rate in the Syn2Real database with the help of SVM classifier and in the case of selecting the proposed feature, it indicates its superiority. The obtained number 100 is a positive predictive rate in the VisDA-2018 database with the help of KNN classifier and in the case of selecting the Feature selection of Laplace monitoring and feature selection of Laplace monitoring in the absence of feature selection, it indicates its superiority. This classifier on the VisDA-2018 database has improved in the feature selection state of the proposed deep learning reinforcement in comparison with the without feature selection state. There are also acceptable results on the Syn2Real database in the feature selection state of the proposed deep learning reinforcement with the help of KNN classifier.

4) The Evaluation of Negative Predictive Rate

The negative prediction rate of the classification was evaluated with the help of three classifiers RF, KNN and SVM on the two databases used in the paper in three states of selecting the Laplace monitoring feature, selecting the proposed deep reinforcement learning feature and without the feature selection. Table 5 demonstrates the negative predictive rate results.

Table 5- The evaluation results of negative predictive rate criterion

The used database	VisDA-2018			Syn2Real		
The used classifier	RF	KNN	SVM	RF	KNN	SVM
Feature selection of the proposed deep reinforcement learning	69.38	95.1	89.12	78.01	72.29	100
Feature selection of Laplace monitoring	63.63	100	90.90	72.11	74.74	90.1
Without feature selection	72.72	44.44	84.61	80	75.48	87.29

The negative predictive rate in the Syn2Real database in the state of feature selection of the proposed deep reinforcement learning was 100%. This showed its superiority in comparison with 90.1% in the state of selecting the Laplace monitoring feature and 87.29 in the case without feature selection with the help of SVM classifier. The obtained number 74.74 is a negative predictive rate in the Syn2Real database with the help of KNN classifier and in the case of selecting the Feature selection of Laplace monitoring and in the state of Without feature selection was 75.48%, KNN indicates its superiority from the state of Feature selection of Laplace monitoring and Feature selection of the proposed deep reinforcement learning.

5) Geometric Mean

The geometric mean of Gmean classification was evaluated with the help of three classifiers RF, KNN and SVM on the two databases used in the paper in three states of selecting the Laplace supervisory feature, selecting the proposed deep reinforcement learning feature and without selecting the evaluated feature. Table 6 demonstrates the obtained Gmean results.

Table 6- Geometric Mean Evaluation Criteria

The used database	VisDA-2018			Syn2Real		
The used classifier	RF	KNN	SVM	RF	KNN	SVM
Feature selection of the proposed deep reinforcement learning	79.47	63.15	81.80	99.5	84.73	84.82
Feature selection of Laplace monitoring	94.23	68.41	78.94	98.25	71.67	81.25
Without feature selection	84.21	73.68	68.25	97.41	75.77	86.87

The obtained number 99.5 is a Gmean results in the Syn2Real database with the help of RF classifier and in the case of feature selection of the proposed deep reinforcement learning, it indicates its superiority. KNN classifier on the Syn2Real database has improved in the feature selection state of the proposed deep learning reinforcement in comparison to the without feature selection state. There are also acceptable results on the VisDA-2018 database in the feature selection state of the proposed deep learning reinforcement with the help of SVM classifier.

6. Conclusion

This paper presents a method for selecting a feature based on deep reinforcement learning in domain adaptation. By comparing the classification methods, it can be seen that SVM has the highest accuracy among the methods. The lowest accuracy is also related to the KNN method. Using the proper feature selection with the help of the proposed method based on deep reinforcement learning and its improvement, has been able to solve the problems and challenges of the classification in the domain adaptation. The use of a proper method in feature selection for classification in domains adaptation enhances the efficiency and evaluation criteria. To evaluate the criteria of accuracy, sensitivity, positive predictive rate as well as negative, predictive rate and Gmean were used. From the obtained results, the superiority of the proposed method can be understood.

7. Statements on compliance with ethical standards and standards of research involving animals

"This article does not contain any studies involving animals performed by any of the authors."

References

[1] Luján-García, J. E., Yáñez-Márquez, C., Villuendas-Rey, Y., & Camacho-Nieto, O. "A transfer learning method for pneumonia classification and visualization". *Applied Sciences*, vol. 10, no. 8, pp. 2908, 2020.

[2] Kenthapadi, K., Mironov, I., &Thakurta, A. G. Privacy-preserving data mining in industry. In Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, pp. 840-841, 2019.

[3] Ramírez-Gallego, S., Mouriño-Talín, H., Martínez-Rego, D., Bolón-Canedo, V., Benítez, J. M., Alonso-Betanzos, A., & Herrera, F. "An information theory-based feature selection framework for big data under apache spark". *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 48, no. 9, pp. 1441-1453, 2017.

[4] Urbanowicz, R. J., Meeker, M., La Cava, W., Olson, R. S., & Moore, J. H. "Relief-based feature selection: Introduction and review". *Journal of biomedical informatics*, vol. 85, pp.189-203, 2018.

[5] Li, J., Cheng, K., Wang, S., Morstatter, F., Trevino, R. P., Tang, J., & Liu, H. "Feature selection: A data perspective". *ACM Computing Surveys* (CSUR)), vol. 50, no. 6, pp.1-45, 2017.

[6] Abualigah, L. M., Khader, A. T., &Hanandeh, E. S. "A new feature selection method to improve the document clustering using particle swarm optimization algorithm". *Journal of Computational Science*, vol. 25, pp. 456-466., 2018.

[7] Hayder A. Alatabi and Ayad R. Abbas. "Sentiment Analysis in Social Media using Machine Learning Techniques". *Iraqi Journal of Science*, vol. 61, no. 1, pp: 193-201 DOI: 10.24996/ ijs. 2020.61.1.22, 2020.

[8] Smadi, S., Aslam, N., & Zhang, L. "Detection of online phishing email using dynamic evolving neural network based on reinforcement learning". *Decision Support Systems*, vol. 107, pp. 88-102, 2018.

[9] Zahraa Naser Shahweli. "Deep Belief Network for Predicting the Predisposition to Lung Cancer in TP53 Gene". *Iraqi Journal of Science*, vol. 61, no. 1, pp: 171-177 DOI: 10.24996 /ijs. 2020.61.1.19, 2020.

[10] Li, J., Zhao, J., & Lu, K. "Joint Feature Selection and Structure Preservation for Domain Adaptation". *In IjCAI*, pp. 1697-1703, 2016.

[11] Sun, F., Wu, H., Luo, Z., Gu, W., Yan, Y., & Du, Q. "Informative feature selection for domain adaptation". *IEEE Access*, vol. 7, pp. 142551-142563, 2019.

[12] Pan, S. J., & Yang, Q. "A survey on transfer learning. IEEE Transactions on knowledge and data engineering", vol. 22, no. 1, pp. 1345-1359, 2009.

[13] Kouw, W. M., &Loog, M. "An introduction to domain adaptation and transfer learning". arXiv preprint arXiv:1812.11806, 2018.

[14] Gautheron, L., Redko, I., &Lartizien, C. Feature selection for unsupervised domain adaptation using optimal transport. In Joint European Conference on Machine Learning and Knowledge Discovery in Databases, pp. 759-776, 2018.

[15] Tahmoresnezhad, J., &Hashemi, S. "Visual domain adaptation via transfer feature learning". *Knowledge and Information Systems*, vol. 50, no. 2 pp. 585-605, 2017.

[16] Zohrizadeh, F., Kheirandishfard, M., &Kamangar, F. Class Subset Selection for Partial Domain Adaptation. In CVPR Workshops, 2019.

[17] Hancer, E., Xue, B., & Zhang, M. "A survey on feature selection approaches for clustering". *Artificial Intelligence Review*, vol. 53, no. 6, pp. 4519-4545, 2020.

[18] Saranya, G., & Pravin, A. Feature Selection Techniques for Disease Diagnosis System: A Survey. In Artificial Intelligence Techniques for Advanced Computing Applications: Springer, pp. 249-258, 2020.

[19] Wang, L., Wang, Y., & Chang, Q. "Feature selection methods for big data bioinformatics: A survey from the search perspective". *Methods*, vol. 111, pp. 21-31, 2016.

[20] Hu, X., Zhou, P., Li, P., Wang, J., & Wu, X. "A survey on online feature selection with streaming features". *Frontiers of Computer Science*, vol. 12, no. 3, pp. 479-493, 2018.

[21] Hessel, M., Modayil, J., Van Hasselt, H., Schaul, T., Ostrovski, G., Dabney, W., ... & Silver, D. Rainbow: Combining improvements in deep reinforcement learning. In Proceedings of the AAAI Conference on Artificial Intelligence, vol. 32, no. 1, 2018.

[22] Krizhevsky, A., & Hinton, G. Learning multiple layers of features from tiny images, Technical Report, University of Toronto, Toronto, Ontario 2009.

[23] Janisch, J., Pevný, T., &Lisý, V. Classification with costly features using deep reinforcement learning. In Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, pp. 3959-3966, 2019.

[24] Pandian, A. Pasumpon, Ram Palanisamy, and KlimisNtalianis, eds. Proceeding of the International Conference on Computer Networks, Big Data and IoT (ICCBI-2019). Vol. 49. Springer Nature, 2020.

[25] V. Mnih et al., "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, p. 529, 2015.

[26] Li, Shoushan, Guodong Zhou, Zhongqing Wang, Sophia Yat Mei Lee, and Rangyang Wang. "Imbalanced sentiment classification." In Proceedings of the 20th ACM international conference on Information and knowledge management, pp. 2469-2472. 2011.

[27] Peng, X., Usman, B., Kaushik, N., Wang, D., Hoffman, J., &Saenko, K. Visda: A synthetic-to-real benchmark for visual domain adaptation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (pp. 2021-2026), 2018.

[28] Peng, X., Usman, B., Saito, K., Kaushik, N., Hoffman, J., &Saenko, K. Syn2real: A new benchmark forsynthetic-to-real visual domain adaptation. arXiv preprint arXiv:1806.09755, 2018.

[29] Razzaghi, Parvin. "Self-taught support vector machines." *Knowledge and Information Systems* 59, no. 3, pp. 685-709, 2019.

[30] Chen, I-Ting, and Hsuan-Tien Lin. "Improving Unsupervised Domain Adaptation with Representative Selection Techniques." In IAL@ PKDD/ECML, pp. 5-21. 2020.

[31] Ryu, Jongbin, Jiun Bae, and Jongwoo Lim. "Collaborative training of balanced random forests for open set domain adaptation." arXiv preprint arXiv:2002.03642 (2020).