Transfer Learning in Hierarchical Feature Spaces

Hua Zuo, Guangquan Zhang, Vahid Behood, Jie Lu
Centre for QCIS
Faculty of Engineering and Information Technology
University of Technology Sydney, Australia
Hua.Zuo@student.uts.edu.au
Guangquan.Zhang, Vahid.Behbood
Jie.Lu@uts.edu.au

Xianli Meng
College of Mathematics and Information Science
Hebei University
Baoding, Hebei, China
mengxl@hbu.edu.cn

Abstract— **Transfer learning provides an approach to solve target tasks more quickly and effectively by using previously acquired knowledge learned from source tasks. As one category of transfer learning approaches, feature-based transfer learning approaches aim to find a latent feature space shared between source and target domains. The issue is that the sole feature space can't exploit the relationship of source domain and target domain fully. To deal with this issue, this paper proposes a transfer learning method that uses deep learning to extract hierarchical feature spaces, so knowledge of source domain can be exploited and transferred in multiple feature spaces with different levels of abstraction. In the experiment, the effectiveness of transfer learning in multiple feature spaces is compared and this can help us find the optimal feature space for transfer learning.**

Keywords-transfer learning, deep learning, feature extraction

I. INTRUDUCTION

Although machine learning technologies have made great achievements in many research areas, most of these technologies work under assumption that source domain and target domain have the same feature space and distribution. It means that if the feature space or/and distribution of the target data change, the prediction model trained using source data can't be used for the target tasks, so new model should be built using adequate labeled target data, which is time-consuming and sometimes unavailable. In real world situations, very few labeled target data can be obtained, and collecting new labeled data and constructing a new prediction model for target tasks is impossible. If knowledge exploited from similar but not identical source domain with plenty of labeled data can be utilized to target tasks, building a well prediction model for target task becomes possible.

Transfer learning has emerged as a way of exploiting knowledge from source domain to improve the performance of target tasks. Unlike traditional machine learning, transfer learning considers source domain and target domain are different. There are many techniques and methods are proposed to transfer knowledge from source domain to target domain. The existing transfer learning approaches can be mainly divided into four categories: Instance-based approaches, Feature-based approaches, Parameter-based approach and relational approaches. Different approaches are suitable for specific scenarios. In Instance-based transfer learning approaches, the general assumption is that source and target domains have a lot of overlapping features [1]. When source and target domains only have some overlapping features, and lots of features only have support in either the source or the target domain, feature-based transfer learning approaches are used to learn a latent feature space that can be shared between source and target domains [2]. The motivation of Parameter-based transfer learning approaches is that the parameters in a well-trained model have learned a lot of structure. If two tasks are related, this structure can be transferred to lean the parameters [3]. And in relational transfer learning approaches, if two relational domains are related, they may share some similar relations among objects. These relations can be used for knowledge transfer across domains [4]. Fuzzy system has been introduced to build model for transfer learning problems. A fuzzy bridge refinement-based domain adaptation method based on fuzzy system and similarity concepts is developed to modify the target instances' labels which are predicted by the prediction model trained using source data [5].

Though these approaches have been introduced as a possible solution for the transfer learning problems, its performance is not yet acceptable. One reason for this is most of the existing approaches only transfer knowledge in one feature space, so all aspects of source knowledge can't be exploited and transferred to target domain. Deep learning provides us the approach to learn feature spaces in a hierarchical structure, so the knowledge of source domain is exploited in different levels of abstraction and different knowledge of source domain can be transferred to solve the tasks in target domain. In addition, in many approaches for transfer learning, target domain are induced to be closer and similar with source domain so that knowledge of source domain can be transferred to target domain. However, this may lead to the loss of information in target domain. Our final aim is to solve tasks in target domain, so a more reasonable way is first extracting information of target domain, and then basing this information to exploit the related knowledge from source domain to help build the model for target domain.

In this paper, we propose a transfer learning method that can exploit feature spaces with a hierarchical structure using deep learning method, so knowledge of source domain can be exploited and transferred in multiple levels of abstraction. In addition, the way of extracting feature spaces can make sure that the information in target domain won't lost and the knowledge extracted from source domain is related with target domain.

The main contributions of this paper are:

(1) It solves the transfer learning problem in the hierarchical feature spaces, so knowledge of source domain with different levels of abstraction can be transferred to target domain.

(2) It extracts feature spaces from target domain to make sure the knowledge exploited from source domain is related to the target domain.

This paper is organized in the following way. In Section II, we start with the literature review of feature-based transfer learning approaches, and deep learning methods that are applied in transfer learning. The proposed transfer learning method with more details is given in Section III. The experiments in Section IV verify the effectiveness of the proposed method. Finally, conclusion and future work are given in Section V.

II. LITERATURE REVIEW

There are four main approaches to solve transfer learning problems, and here we only focus on the feature representation transfer learning approach which aims to learn a 'good' feature representation for target domain. In this case, the knowledge used to transfer across domains is encoded into the learned feature representation. With the new feature representation, the performance of the target task is expected to improve significantly.

In the prediction problem, how to choose appropriate features is important to achieve a good performance. When selecting the features, many algorithms consider all the features are equal and relevant. Baralis et al. present a method for learning a low-dimensional representation which is shared across a set of multiple related tasks. Their method learns a few features common across the tasks by regularizing within the tasks while keeping them coupled to each other. Moreover, the method can be used, as a special case, to select a few features from a prescribed set [6]. Raina et al. pose the self-learning problems mainly to formalize a machine learning framework that has the potential to make learning significantly easier and cheaper. Based on the assumption that the unlabelled data can be assigned to the supervised learning task's class labels, they present largely unsupervised learning algorithms for improving performance on supervised classification tasks [7]. Dai et al. address a text-mining task, where the labelled data are under one distribution in one domain known as in-domain data, while the unlabelled data are under a related but different domain known as out-of-domain data. They propose a novel co-clustering based classification algorithm to solve this problem. A key intuition of their work is that even though the two domains may be under different distributions, they are able to identify a common part between them [8].

Deep learning is an emerging research area and its prominent characteristic is multiple hidden layers, which can capture the intricate non-linear representation of data. Deep learning methods are based on deep neural network that can learn data representation with different levels of abstraction, and the high-level representation is learned on a basis of low-level representation. Actually, the representations in each layer can be regarded as a feature space, and the every representation is a feature in the feature space.

The multi-level structure is first proposed by Hubel and Wiesel [9], and the multi-stage Hubel-Wiesel architectures consist of alternating layers of convolutions and max pooling to extract new representation for data. Ahmed et al. use the Hubel-Wiesel architectures to solve the multiple tasks problems [10]. In their method, the tasks in target domain and related domain are trained together to get one neural network, so the target domain and related domain share the same input and hidden layers, but each task has their separate output neurons. The above method only focus on the case that each task only has one output, then Huang et al. improve the above method to solve the scenario that every task has multiple outputs, such as multiple category classification problems [11]. Instead of sharing hidden layers between target domain and related domains, whether shared hidden layers trained by the source task can be reused on a different target task is detected. For the target task model, only the last classification layer needs to be retrained, but any layer of the new model could be fine-tuned if desired. In this case, the parameters of hidden layers in the source task model act as initialization parameters of the new target task model, and this strategy is especially promising for a model in which good initialization is very important [12]. Different with these deep learning structures in which all layers except the output layer are used to learn feature space for data, Collobert et al. propose a deep learning structure [13]. The first two layers are used to extract features at different levels, such as word level and sentence level in Natural Language Processing, and subsequent layers are classical neural network layers used for prediction. Except the Hubel-Wiesel architecture, Stacked Denoisng Autoencoder (SDA) is another important structure in deep learning. The training process of SDA includes two steps: the first step is an unsupervised pre-training process of all hidden layers, and the second step is a fine-tuned process in a supervised learning way [14]. Based on the SDA model, different feature transference strategies are introduced to target tasks with varying degrees of complexity. The number of layers transferred to the new model depends on the high-level or low-level feature representations that are needed. This means if low-level features are needed, only the first layer parameters are transferred to the target task [15]. In addition, based on the deep neural network structure, interpolating path method is used to transfer knowledge from source domain to target domain. On a basis of the Grassman manifold, the high-dimension feature spaces of source domain and target domain are projected to a feature space with low dimension. This method presents a way to interpolate smoothly between source and target domains, a series of feature sets is generated on the interpolating path and intermediate feature extractors are formed based on deep neural network [16].

III. TRANSFER LEARNING MODEL BASED ON HIERARCHICAL FEATURE SPACES

In this section, the proposed transfer learning method is described with more details. Before giving the concrete steps of our proposed method, some related concepts in transfer learning and the notations used through this section will be given.

Definition 1 (Domain) [17] A domain, which is denoted by $D = \{ c, P(X)\}$ consists of two components:

(1) Feature space χ ; and

(2) Marginal probability distribution $P(X)$ where $X = \{x_1, \cdots, x_n\} \in \chi$.

Definition 2 (Task) [17] A task, which is denoted by $T = \{Y, f(\cdot)\}$, consists of two components:

(1) A label space $Y = \{y_1, \cdots, y_m\}$; and

(2) An objective predictive function $f(\cdot)$ which is not observed and is to be learned by pairs $\{x_i, y_i\}$.

Definition 3 (Transfer learning) [17] Given a source domain D_s and learning task T_s , target domain D_t and learning task T_t , transfer learning aims to improve the learning of the target predictive function $f_t(\cdot)$ in D_t using the knowledge in D_s and T_s where $D_s \neq D_t$ or $T_s \neq T_t$.

In the above definition, the condition $D_s \neq D_t$ implies that either $\chi_s \neq \chi_t$ or $P_s(X) \neq P_t(X)$. Similarly, the condition $T_s \neq T_t$ implies that either $Y_s \neq Y_t$ or $f_s(\cdot) \neq f_t(\cdot)$.

In addition, there are some explicit or implicit relationships between the feature spaces of two domains such that we imply that the source domain and target domain are related. It should be mentioned that when the target and source domains are the same ($D_s = D_t$) and their learning tasks are also the same ($T_s = T_t$) the learning problem becomes a traditional machine learning problem.

In our method, SDA is used to learn hierarchical feature spaces. The elementary unit in SDA is Denoising Autoencoders. Next, the structures of Autoencoders and Denoising Autoencoders are given as follows. The structure of Autoencoders is designed to reconstruct the input data. The structure of Autoencoders is shown in Figure 1:

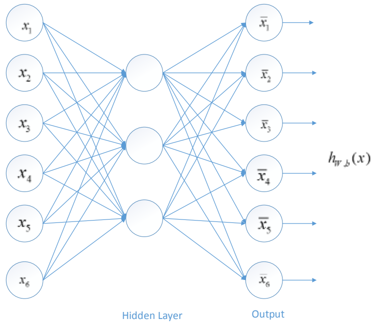

Figure 1: Autoencoders

The reconstruction accuracy is obtained by minimizing the average reconstruction error between the original data and the reconstructed instances. The neurons in the hidden layer have the ability of reconstructing the input data, so they can be treated as a new feature space for the original data. The training process of Antoencoders is to minimize the below formula:

$$\frac{1}{m}\sum_{i=1}^{m} g(x^{(i)}, h_{W,b}(x^{(i)})) \quad (1)$$

where $g(x^{(i)}, h_{W,b}(x^{(i)}))$ is the distance between the input data and the reconstructed data.

But this reconstruction criterion alone may lead to the obvious solution, which simply copies the input. In view of this situation, Pascal [14] gave the definition of a good representation and followed it as new criterion to reconstruct. They defined "a good representation is the one that can be obtained robustly from a corrupted input and that will be useful for recovering the corresponding clean input". So in order to extract new features that are stable and robust under corruptions of the input, denoising is advocated as a training criterion in Antoencoders to extract features capture useful structure in the input distribution. First all the data will be added with some noise, so the original data $x^{(i)}$ becomes $\bar{x}^{(i)}$, and the function needed to be optimized becomes:

$$\frac{1}{m}\sum_{i=1}^{m} g(x^{(i)}, h_{W,b}(\bar{x}^{(i)})) \quad (2)$$

SDA is stacking many Denoising Autoencoders together. SDA consists of layers of Denoising Autoencoders in which the outputs of each layer are wired to the inputs of the successive layer.

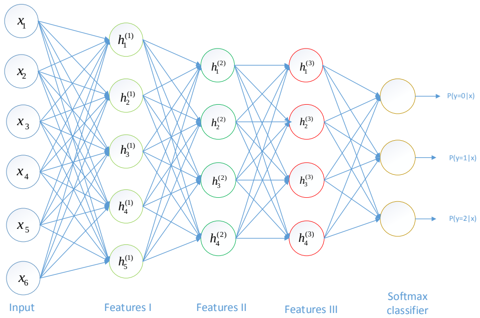

Figure 2: Stacked Denosing Autoencoders

The above figure gives a SDA structure with three hidden layers. The training process of this deep neural network includes two steps. In the first step, the parameters between the input and the hidden layers are trained in an unsupervised way, so in this step, only unlabelled data are used to pre-train the network. In the second step, labelled data are used to fine-tune the network in a supervised learning way, so the parameters between the input layer and the hidden layer are fine-tuned and the parameters between the hidden layers and the output layer are trained. The hidden layers of the network can be regarded as learned feature spaces with different levels of abstraction, and the low-level feature space is learned on a basis of high-level feature space. In our method, deep learning structure is used to extracted feature spaces, thus the softmax classifier layer can be deleted. The modified SDA has the structure in Figure 3:

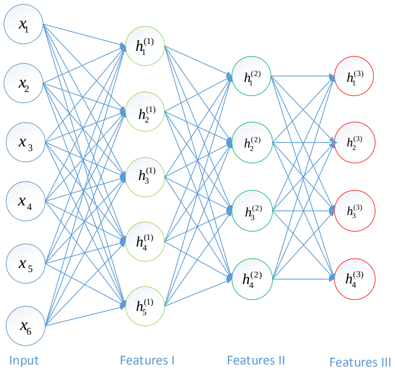

Figure 3: Revised Stacked Denoising Autoencoders

Next, the specific steps of the proposed transfer learning method are as follows:

Step 1: Learn feature spaces with hierarchical structure.

Our final aim is to solve the tasks in target domain, so the knowledge exploited form source domain should be related with target domain. In order to guarantee the correlation of the exploited knowledge and target domain, unlabelled target data are used to extract the feature spaces to make sure the feature spaces are restricted by target domain.

Unlabelled target data are used to train the deep neural network with the structure in Figure 3. As shown in Figure 3, there are three hidden layers in the neural network, thus three feature spaces can be learned, denoted as χ_1, χ_2, χ_3.

Step 2: Extract knowledge from source domain in every feature space.

First, source data are projected to the new feature spaces, so the marginal probability distribution of source data changes in the new feature space. Therefore, after the transformation of feature space of source data, we have the following changes in source domain:

$$\chi_s \to \chi_j, P_s(X) \to P_j(X), j = 1,2,3 \qquad (3)$$

Source data are projected into the feature space that are extracted from target domain, so the representation of source data in the new feature spaces can be understood as being restricted in the range that is defined by target domain.

Then knowledge of source domain is extracted to help solve the tasks in target domain. Source data with new representation in new feature spaces are used to exploit the knowledge in source domain, and the new feature space that are learned from target data can guarantee the correlation of the extracted knowledge and target domain.

Step 3: Build classifier for target domain in every feature space.

Before training the classifier for target domain, target data are also projected into new feature spaces. Similar with source domain, feature space and marginal probability distribution in target domain have the following changes:

$$\chi_t \to \chi_j, P_t(X) \to P_j(X), j = 1,2,3 \qquad (4)$$

Although the target data are projected into new feature spaces, the way of extracting feature spaces can guarantee minimal loss of information in target domain. Since the feature spaces are learned from target domain, the reconstruction optimization criterion makes sure the most information of target domain are keep in the new representations of data.

Then the labelled target data with new representation are used to train the classifier for target domain. Because the labelled target data are not sufficient enough to train a well classifier, the knowledge exploited from source domain are used to help build the classifier. Since the knowledge of source domain is contained in the parameters of the classifier trained in Step 2, we utilize the knowledge by initialize the parameters of the new classifier with the parameters of the classifier trained in Step 2, and then the parameters are fine-tuned by the labelled target data with new representation.

After the training process, the classifier for target domain is used to predict the labels of unlabelled data in target domain. Because different knowledge can be learned from different feature spaces with different levels of abstraction to help improve the performance of classifier for target domain, the accuracies of the classifier trained in different feature spaces are also different. The level of abstraction increases as the increase of the layers, and we want to exploit the relationship of the level of abstraction of a layer and the accuracy of the classifier built in the feature space extracted from that layer, and this can provide us information to choose the optimal feature space to transfer knowledge.

IV. EXPERIMENTS

Experiment is designed to verify the effectiveness of the proposed transfer learning method and exploit the relationship of the hierarchical feature spaces and the accuracies of the classifiers built in these feature spaces.

(a) Dataset

Two public image datasets MNIST and MADbase are used in the experiments. MNIST and MADbase are Arabic handwritten digits with different forms, where MNIST is Western Arabic numerals and MADbase is Eastern Arabic numerals respectively. The images in these two datasets are size-normalized to 28×28 (784) pixels. In our experiments, 8000 labelled samples are selected from MNIST as source domain. 6000 samples are selected from MADbase as target domain, but only 1000 of them are labelled.

(b) Experiment results

In this experiment, five feature spaces are extracted, and the dimensions of the feature spaces are listed in Table I. Actually, the dimensions of the feature spaces are the number of neurons in the hidden layers. Based on the feature spaces extracted from target domain, related knowledge from source domain are exploited and utilized to help build the classifier for target tasks. The accuracies of the classifiers in five feature spaces are also listed in Table I.

TABLE I. THE DIMENSIONS OF THE FEATURE SPACES

Feature Space	Dimension	Accuracy %
0	784	89.03
1	700	93.61
2	600	93.44
3	500	92.86
4	400	91.92
5	300	92.99

Feature space 0 is the original feature space of target domain, and the dimension of the original feature space is 784. From Table I we can see that the accuracy of the classifier built in the original feature space is 89.03%, and the accuracies of the classifiers built in the extracted five feature space are all greater than 91%. This indicates the effectiveness of the proposed granular transfer learning method.

In order to compare the accuracies of the classifier in the five extracted feature space, the tendency is shown in Figure 4.

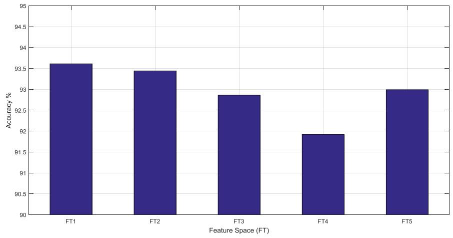

Figure 4: Accuracies of the classifiers in five feature spaces

From Figure 4 we can see that, in the five feature spaces, the highest accuracy is 93.61% in the first feature space, and the lowest accuracy is 91.92% in the fourth feature space. With the increase of the level of granularity, the accuracy of the classifier built in the corresponding feature space first declines and gets to the bottom in the fourth feature space, and then rises in the fifth feature space.

V. CONCLUSION AND FUTURE WORK

In this paper, we propose a transfer learning method to exploit and transfer knowledge in multiple feature spaces with different levels of abstraction. From the results of the experiments, we can see that different knowledge can be exploited from feature spaces with different levels of abstraction, so the accuracies of the classifier in these feature spaces are also different. In our experiment, the numbers of neurons in the hidden layers are given, and the structure of the deep neural network is specific. In the future study, the impact of the structure of deep neural network to the hierarchical feature spaces will be considered.

REFERENCES

[1] Kanamori, T., Hido, S. & Sugiyama, M. 2009, 'A least-squares approach to direct importance estimation', *The Journal of Machine Learning Research*, vol. 10, pp. 1391-445.

[2] Glorot, X., Bordes, A. & Bengio, Y. 2011, 'Domain adaptation for large-scale sentiment classification: A deep learning approach', *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pp. 513-20.

[3] Evgeniou, T. & Pontil, M. 2004, 'Regularized multi--task learning', *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, pp. 109-17.

[4] Li, F., Pan, S.J., Jin, O., Yang, Q. & Zhu, X. 2012, 'Cross-domain co-extraction of sentiment and topic lexicons', *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, Association for Computational Linguistics, pp. 410-9.

[5] Behbood, V., Lu, J. & Zhang, G. 2014, 'Fuzzy refinement domain adaptation for long term prediction in banking ecosystem', *IEEE Transactions on Industrial Informatics*, vol. 10, no. 2, pp. 1637-46.

[6] Baralis, E., Chiusano, S. & Garza, P. 2008, 'A lazy approach to associative classification', *IEEE Transactions on Knowledge and Data Engineering* , vol. 20, no. 2, pp. 156-71.

[7] Raina, R., Battle, A., Lee, H., Packer, B. & Ng, A.Y. 2007, 'Self-taught learning: transfer learning from unlabeled data', *Proceedings of the 24th international conference on Machine learning*, ACM, pp. 759-66.

[8] Dai, W., Xue, G.-R., Yang, Q. & Yu, Y. 2007, 'Co-clustering based classification for out-of-domain documents', *Proceedings of the 13th*

ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, pp. 210-9.

[9] Hubel, D.H. & Wiesel, T.N. 1962, 'Receptive fields, binocular interaction and functional architecture in the cat's visual cortex', *The Journal of physiology*, vol. 160, no. 1, pp. 106-54.

[10] Ahmed, A., Yu, K., Xu, W., Gong, Y. & Xing, E. 2008, 'Training hierarchical feed-forward visual recognition models using transfer learning from pseudo-tasks', *Computer Vision–ECCV 2008*, Springer, pp. 69-82.

[11] Huang, J.-T., Li, J., Yu, D., Deng, L. & Gong, Y. 2013, 'Cross-language knowledge transfer using multilingual deep neural network with shared hidden layers', *2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* , IEEE, pp. 7304-8.

[12] Cireşan, D.C., Meier, U. & Schmidhuber, J. 2012, 'Transfer learning for Latin and Chinese characters with deep neural networks', *The 2012 International Joint Conference on Neural Networks (IJCNN)* , IEEE, pp. 1-6.

[13] Collobert, R. & Weston, J. 2008, 'A unified architecture for natural language processing: Deep neural networks with multitask learning', *Proceedings of the 25th international conference on Machine learning*, ACM, pp. 160-7.

[14] Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y. & Manzagol, P.-A. 2010, 'Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion', *The Journal of Machine Learning Research*, vol. 11, pp. 3371-408.

[15] Kandaswamy, C., Silva, L.M., Alexandre, L.A., Santos, J.M. & de Sá, J.M. 2014, 'Improving deep neural network performance by reusing features trained with transductive transference', *Artificial Neural Networks and Machine Learning–ICANN 2014*, Springer, pp. 265-72.

[16] Chopra, S., Balakrishnan, S. & Gopalan, R. 2013, 'Dlid: Deep learning for domain adaptation by interpolating between domains', *ICML workshop on challenges in representation learning*, vol. 2, p. 5.

[17] Pan, S.J. & Yang, Q. 2010, 'A survey on transfer learning', *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345-59.