Neural Networks 32 (2012) 174–178

Contents lists available at SciVerse ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

2012 Special Issue

Analysis of the IJCNN 2011 UTL challenge

Isabelle Guyon [a,*], Gideon Dror [b], Vincent Lemaire [c], Daniel L. Silver [d], Graham Taylor [e], David W. Aha [f]

[a] *Clopinet, CA, USA*
[b] *Yahoo!, Haifa, Israel*
[c] *Orange Labs, France*
[d] *Acadia University, Canada*
[e] *New York University, USA*
[f] *Naval Research Laboratory, USA*

ARTICLE INFO

Keywords:
Machine learning
Transfer learning
Unsupervised learning
Metric learning
Kernel learning
Unlabeled data
Challenge
Competition

ABSTRACT

We organized a challenge in "Unsupervised and Transfer Learning": the UTL challenge (http://clopinet.com/ul). We made available large datasets from various application domains: handwriting recognition, image recognition, video processing, text processing, and ecology. The goal was to learn data representations that capture regularities of an input space for re-use across tasks. The representations were evaluated on supervised learning "target tasks" unknown to the participants. The first phase of the challenge was dedicated to "unsupervised transfer learning" (the competitors were given only unlabeled data). The second phase was dedicated to "cross-task transfer learning" (the competitors were provided with a limited amount of labeled data from "source tasks", distinct from the "target tasks"). The analysis indicates that learned data representations yield significantly better results than those obtained with original data or data preprocessed with standard normalizations and functional transforms.

1. Introduction

Classical machine learning techniques, including artificial neural networks, work well if data are independently and identically distributed and if there there are sufficient numbers of labeled training data. Unfortunately, in many real world applications, these assumptions are violated: (1) Data available for training are not always similar to data the system will be exposed to when it is deployed. (2) Few labeled training data may be available due to the cost or burden of manually annotating data. Transfer learning addresses both of these shortcomings by providing tools to learn a new task for which labeled data are scarce from a related task for which data are abundant, labeled or not (Fig. 1).

2. Definition of unsupervised and transfer learning

In this paper, we call "domain" the input space (*e.g.,* a feature vector space) and we call "task" the output space (represented by labels for classification problems). We use the adjective "source" for an auxiliary problem, for which we have an abundance of data (*e.g.*, classifying house cats), and "target" for the problem of interest (*e.g.,* classifying tigers). A transfer learning system recognizes and applies knowledge and skills learned on a "source" problem to a novel "target" problem. Within this framework, there several settings, depending on whether *labels are available for the source task and/or the target task*, and *whether domains and tasks are the same or different for the source and the target problem* Pan and Yang (2010). Some familiar settings include:

Inductive transfer learning: Labeled data are available for the target tasks to perform supervised learning, but not in abundance. A typical setting is that of a large multi-class problem, in which some classes are a lot more depleted than others (*e.g.,* in image classification or text classification). The most abundant classes serve as "source tasks" for the least abundant ones, representing the "target tasks". If all source data are labeled, one talks of *multi-task learning*. When unlabeled source data are available, the term *self-taught learning* Raina, Battle, Lee, Packer, and Ng (2007) is used.

Domain Adaptation: Labeled data are available for both the source and the target tasks, but the input distribution (domain) is changing. For instance a speech recognition system would be trained with native speakers and be used by a non-native speaker willing to partially re-train it.

Semi-supervised and transductive learning: There is no change in either domain or task and labeled training data are available. In addition, either unlabeled training data are available (*semi-supervised learning*) or the unlabeled test data may be used for learning (*transductive learning*). An example of semi-supervised

* Correspondence to: Clopinet, 955 Creston Road, Berkeley, CA 94708, USA. Tel.: +1 510 524 6211; fax: +1 510 524 6211.
E-mail address: guyon@clopinet.com (I. Guyon).

0893-6080/$ – see front matter © 2012 Elsevier Ltd. All rights reserved.
doi:10.1016/j.neunet.2012.02.010

Fig. 1. The transfer learning question: Can learning about house cats help us learn about tigers?

learning would be classifying images of skin lesions into cancer and non-cancer from a database of examples including a few confirmed cases.

Unsupervised learning and cross-task transfer learning: When no labeled data are available at all (in the source or target domain), *unsupervised learning* techniques (clustering, vector quantization, factor analysis, manifold learning, etc.) may be applied in an effort to represent data in a simpler and/or better way, either for visualization, or as data preprocessing, for future use in supervised learning tasks. When a limited amount of labeled data is available *only* in the source domain, we talk about *cross-task transfer learning*.

3. Setting and tasks of the challenge

The case examined in this challenge was that of Unsupervised and Cross-Task Transfer Learning. Labels were available for target tasks to the challenge organizers only, to evaluate data representations provided by the participants. This setting may seem artificial because there are ultimately labels available for some target tasks, so why not make same available to the participants? Our goal was to decouple problems and see how far one could get by working only on learning data representations. This exercise has turned out to be extremely fruitful.

The datasets of the challenge (Table 1) were split into a large development set,[1] a validation set and a final evaluation set. *The goal of the challenge was to produce good data representations on the final evaluation set.* The validation set is similar to the final evaluation set; it was provided for practice. The assessment of the data representations was carried out on *target tasks* (that are supervised learning classification tasks), using labels known only to the competition organizers. The target tasks for the validation set and the final evaluation set are different but related. The intention is to determine the extent to which the abstract features are useful for classifying a family of related tasks. During the development period, online feedback was provided only on the validation set. The results on the final evaluation set were revealed only at the end of the challenge.

The challenge proceeded in two phases. The first phase focused on *unsupervised learning*.[2] During that phase, no labels were provided to the participants in either the source or the target domain. It was then followed by a second phase on *cross-task transfer learning* for which some labels for *source tasks, distinct from the target tasks*, were provided for a subset of the development data.

We selected five different application domains that are illustrative of fields in which transfer learning is applicable[3]:

AVICENNA: Professor Mohamed Chériet, École de Technologie Supérieure, University of Quebec, Montréal, Canada, and his students prepared a large corpus of historical Arabic documents Moghaddam, Cheriet, Adankon, Filonenko, and Wisnovsky (2010). Transfer learning methods could accelerate the application of handwriting recognizers to historical manuscripts by reducing the need for using human experts to label and index them.

HARRY: The identification and recognition of gestures, postures and human behaviors has gained importance in applications such as video surveillance, gaming, marketing, computer interfaces and interpretation of sign languages for the deaf. The HARRY dataset was constructed from the KTH human action recognition dataset of Ivan Laptev and Barbara Caputo[4] and the Hollywood 2 dataset of human actions and scenes of Marcin Marszalek, Ivan Laptev, and Cordelia Schmid.[5]

RITA: Object recognition in images is a classical pattern recognition task gaining importance for retrieval applications, including for Internet search. The RITA dataset was constructed from the CIFAR dataset of Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton,[6] a subset of the 80 million Tiny images dataset of Antonio Torralba, Rob Fergus, and William T. Freeman.[7]

SYLVESTER: Massive datasets need to be processed automatically to assist experts in ecology, geography, geology, climatology, archeology, and seismology. We used data from the US Forest service to illustrate such tasks.[8]

TERRY: Internet search engines process billions of queries daily to rank web pages. Very few labeled data are available, but millions of documents must be indexed. We used a subset of the RCV1-v2 Text Categorization Test Collection derived from Reuter's news articles formatted and made publicly available by Lewis et al. (2004).

These are all multi-class problems. We selected a subset of the classes as target tasks and used the remainder as source tasks. The tasks in the source and target domains are very distinct, for instance, the source domain may include pictures of cars, houses,

[1] We use the nomenclature "development set" rather than "training set" to stress that the actual (supervised) training is performed with labeled target domain data by the organizers only. The development set is not the training set for the target tasks.

[2] From the point of view of the overall problem, including the evaluation on supervised target tasks performed by the organizers, this setting is similar to self-taught learning Raina et al. (2007) because of the availability of unlabeled data in a source domain distinct from the target domain.

[3] A detailed technical report on the datasets was made available after the challenge ended: http://www.causality.inf.ethz.ch/ul_data/DatasetsUTLChallenge.pdf.

[4] http://www.nada.kth.se/cvap/actions/.

[5] http://www.irisa.fr/vista/Equipe/People/Laptev/download.html.

[6] http://www.cs.toronto.edu/~kriz/cifar.html.

[7] http://groups.csail.mit.edu/vision/TinyImages/.

[8] http://archive.ics.uci.edu/ml/datasets/Covertype.

Table 1
UTL challenge datasets. *Devel* = number of examples in development data. *Transf* = number of source task labels released in the second phase. The validation and final evaluation sets consist of 4096 examples each.

Dataset	Domain	Features	Sparsity (%)	Devel.	Transf.
AVICENNA	Handwriting	120	0	150 205	50 000
HARRY	Video	5 000	98.1	69 652	20 000
RITA	Images	7 200	1.1	111 808	24 000
SYLVESTER	Ecology	100	0	572 820	100 000
TERRY	Text	47 236	99.8	217 034	40 000

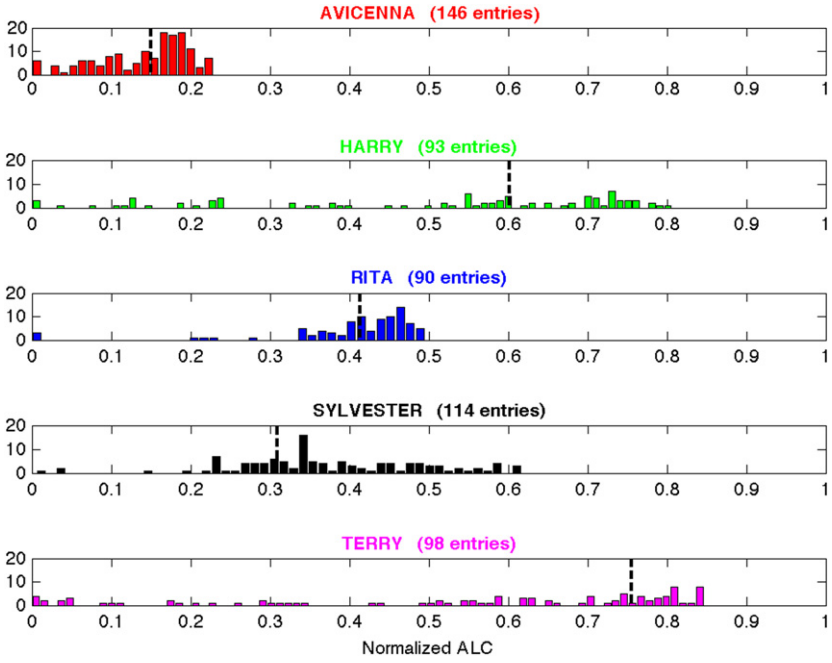

Fig. 2. Histogram of participant results for the first phase of the challenge. The *y* axis is the number of entries in each bin. Similar histograms were obtained in the second phase (not shown).

trees, and the target domain would include horses, chairs, and computers. The datasets varied in difficulty, as illustrated by the histograms of performances of the participants (Fig. 2). AVICENNA is very hard, HARRY and TERRY are the easiest tasks (but there are a very wide range of results) and RITA and SYLVESTER are of medium difficulty.

4. Protocol and evaluation

The challenge protocol was inspired by previous competitions we organized ChaLearn (0000) and was designed to ensure fairness of the evaluation and stimulate participation. We provided guidance to the participants with detailed answers to Frequently Asked Questions (FAQ)[9] and we posted a short video tutorial on transfer learning.[10] The rules can be found on the website of the challenge.[11]

The data representations were assessed automatically on the website of the challenge. The evaluation software and sample code were provided to the participants. The organizers defined several binary classification tasks unknown to the participants and placed them in either the validation set or the final evaluation set. The platform used the data representations provided by the participants to train a linear classifier to solve these tasks. So in all cases, the data representations were assessed with supervised learning tasks. We chose a simple Hebbian learning rule.[12] The Hebbian algorithm has a kernelized version. Thus, we let the participants submit either a data representation or a semi-definite positive matrix (interpreted as a matrix of similarity between pairs of examples), which was used as a kernel. See Guyon, Dror, Lemaire, Taylor, and David W (2011) for details.

To compute the ranking score, a form of cross-validation was performed by partitioning randomly the evaluation data (validation set or final evaluation set) multiple times into a training and a test set, and averaging performances. The number of training examples m was varied from 1 to 2^6 and the area under the ROC curve[13] (AUC) was plotted against m in a log2 scale to emphasize the results on small m. The area under the learning curve (ALC) was used as a scoring metric to assess the results. The ALC criterion is a good way to aggregate the values of the AUC over all the considered number of training examples. The participants were ranked by ALC for each individual dataset. The participants having submitted a complete experiment (*i.e.*, reporting results on all 5 datasets of the challenge) could enter the final ranking. The winner was determined by the best average rank over all datasets for the results on a complete experiment of their choice. See Guyon et al. (2011) for details.

[9] http://www.causality.inf.ethz.ch/unsupervised-learning.php?page=FAQ.
[10] http://www.youtube.com/watch?v=9ChVn3xVNDI.
[11] http://clopinet.com/ul.

[12] Experiments conducted after the challenge using the submissions of the top ranking participants revealed that other linear classifiers, not making an independence assumption between examples, do not yield performance improvements. This can be explained by the fact that it is possible to orthogonalize the examples with *e.g.*, PCA, making Hebbian learning quasi-optimal. Additionally, for small training sets, Hebbian learning is robust against overfitting.
[13] This is the area under the curve plotting hit rate *vs.* false alarm rate.

Table 2
Normalized ALC values of the top ranking participants.

Rank	Team	Experiment	Avicenna	Harry	Rita	Sylvester	Terry
Phase 1—unsupervised learning							
1	AIO	AIO	0.2183 (1)	0.7043 (6)	0.4951 (1)	0.4569 (6)	**0.8465** (1)
2	1055A	exp1	0.1906 (6)	0.7357 (3)	0.4782 (5)	0.5828 (1)	0.8437 (2)
3	Airbus	A3XX	0.2174 (2)	0.7545 (2)	0.4724 (7)	0.4949 (4)	0.8390 (3)
4	LISA	LISA	0.1960 (5)	0.8062 (1)	0.4731 (6)	0.4763 (5)	0.7959 (6)
Phase 2—transfer learning							
1	LISA	agartha	**0.2273** (1)	**0.8619** (1)	**0.5029** (1)	0.5650 (3)	0.8160 (2)
2	tkgw	crush	0.1973 (2)	0.7533 (2)	0.4095 (4)	**0.5933** (1)	0.8118 (3)
3	1055A	phase2exp1	0.1511 (4)	0.7381 (3)	0.4992 (2)	0.5873 (2)	0.8437 (1)
4	FAST	teaf	0.1909 (3)	0.3580 (4)	0.4275 (3)	0.3379 (5)	0.6485 (4)

5. Results

The challenge attracted 76 participants. There was more participation in the first phase than in the second phase: In the first phase, 6933 jobs were submitted, including 41 complete final entries, while, in the second phase, 1141 jobs were submitted including 14 complete final entries. There were in the end 16 ranked teams in the first phase and 8 ranked teams in the second phase. Not all teams decided to enter the final ranking, despite the option to preserve their anonymity.

The results of the top ranking teams are shown in Table 2. The "normalized" ALC refers to $(ALC - Arand)/(Amax - Arand)$, where Amax is the ALC obtained when perfect predictions are made and Arand is (in expectation) the ALC obtained for random predictions. We show in boldface the best result for both phases. The numbers in parentheses are the ranks for the individual datasets in each phase. The complete result tables are available online[14] and details are provided in Guyon et al. (2011). In support of the significance of the results of the challenge, the top ranking teams in both phases used consistently the same principled methods on all datasets, and performed well on all of them.

It was important for us to assess whether unsupervised learning helps compared to classical normalizations or no preprocessing at all. We ran a couple of baseline algorithms for comparison. Using unsupervised learning, the participants outperformed the organizers on 4/5 datasets (for HARRY, the normalized representation achieved the best results in phase 1).

Finally, we examined the correlation between validation set and final evaluation set performances (Fig. 3). The graph reveals that on several datasets the test set was easier that the validation set. We did this on purpose so the participants would not be frustrated. After removing a few outliers (probably due to submission errors) we obtained a correlation coefficient of 0.88 in the first phase and 0.89 in the second phase. Most participants simply used the validation set performance as a model selection criterion. Because of the high correlation between validation performance and test performance, this turned out to be an effective strategy.

We surveyed the participants to determine what algorithms, software and hardware was used and the best entrants were asked to produce a full length paper, which will appear in JMLR W&CP Guyon et al. (in press). We briefly summarize the methods and findings.

In the first phase, the winner (team name: AIO) used an algorithm to train kernels Aiolli (in press). Using validation data, they incrementally improved their kernel, each time checking the performance on the leaderboard. They developed a systematic method of sequential kernel transformations and recorded which

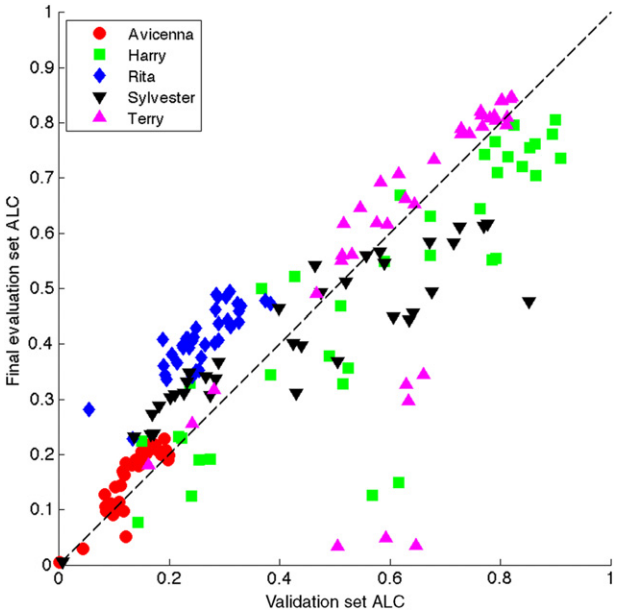

Fig. 3. Correlation between results on the validation and test data in phase 1. A similar plot was obtained with phase 2 data.

sequence ended up giving the best performance on validation data. Then, they applied the same sequence to the final evaluation data. The method can be interpreted as a greedy search for hyperparameters of a compound kernel. The AIO team did not use the development dataset for training at all: The cascade of steps selected with the validation set was then applied to the final evaluation data to produce a new kernel. Thus this method can be qualified of "transductive".

The team LISA, ranking first in the second phase and fourth in the first phase, based their solution on Deep Learning techniques, in particular for unsupervised learning of representations Mesnil et al. (in press). Their methods follow the techniques described in Bengio (2009). Those exploit as building blocks unsupervised learning of single-layer models, such as Restricted Boltzmann Machines, to construct deeper models such as Deep Belief Networks. Their last layer is a transductive PCA (PCA computed directly on the final evaluation data, for the feature representation produced by the preprocessor trained with the development and validation data).

The team 1055A, ranking second in the first phase and third in the second phase used classical unsupervised learning methods: Principal component analysis (PCA) and k-means clustering Liu, Xie, Xiong, and Ge (in press). They first computed the principal components on validation dataset and used the on-line feedback to determine the first n principal components that gave the best global score. Clustering was then performed in the PCA representation and repeated 100 times with different class seeds.

14 Result tables: http://www.causality.inf.ethz.ch/unsupervised-learning.php?page=results.

The number of clusters was optimized with the feedback from the validation dataset. They submitted data representations in a binary encoding of cluster membership.

The team tkgw, ranking second in the second phase, used a method called "Random Forest Proximity", which recursively searches for principal directions when going down a decision tree (halting at a depth of 12). Random Forests Breiman (2001) are ensembles of decision trees built by resampling variables and training examples. The method allowed the authors to create a large number of features, from which they generated a similarity measure. The similarity measure was then turned into a semi-definite positive kernel matrix with a suitable normalization. See the fact sheet for details.

The team Airbus, ranking third in the first phase, tried various preprocessing methods and selected the best one using the validation set. On AVICENNA they ended up using PCA with 90% of variance, and then a RBF kernel. For HARRY and TERRY, they rotated the representation then used a linear kernel. For RITA, they ran k-means to get some clusters then Maximum Variance Unfolding (MVU) on each of them to select features, and then a denoising algorithm. For SYLVESTER, they used whitening, taking 90% of the variance.

The setting of the challenge revealed the power of unsupervised learning as a preprocessor. For all the datasets, unsupervised learning produced results significantly better than the baseline methods (raw data or simple normalizations). The participants exploited effectively the feedback received on the validation set to select the best data representations. The skepticism around the effectiveness of unsupervised learning is justified when no performance on a supervised task is available. However, unsupervised learning can be the object of model selection using a supervised task, similarly to preprocessing, feature selection, and hyperparameter selection. An interesting new outcome of this challenge is that the supervised tasks used for model selection can be distinct from the tasks used for the final evaluation. So, even though the learning algorithms are unsupervised, transfer learning is happening at the model selection level.

In phase 1, there was a danger of overfitting by trying too many methods and relying too heavily on the performance on the validation set. One team for instance overfitted in phase 1, ranking 1st on the validation set, but only 4th on the final evaluation set. Possibly, criteria involving both the reconstruction error and the classification accuracy on the validation tasks may be more effective for model selection. This should be the subject of further research. In phase 2, the participants had available "transfer labels" for a subset of the development data (for classification tasks distinct from the classification tasks of the validation set and the final evaluation set). Therefore, they had the opportunity to use such labels to devise transfer learning strategies. The most effective strategy seems to have been to use the transfer labels for model selection again. None of the participants used those labels for learning.

6. Conclusions

The results of the challenge confirm results reported in the literature that unsupervised learning can be beneficial for preprocessing. The benefits of transfer learning in the "cross-task transfer learning" setting studied in this challenge are mainly derived from an improvement in model selection of unsupervised learning preprocessing techniques. In particular, the challenge demonstrated that the validation data needs not to be drawn from the same distribution, it suffices that the source task used bears some resemblance to the target task. The importance of the degree of resemblance of the two tasks remains to be determined.

Overall, an array of algorithms were used, including classical linear methods like Principal Component Analysis (PCA), and non-linear methods like clustering (k-means and hierarchical clustering being the most popular), and Kernel-PCA (KPCA), as well as cutting edge methods including non-linear auto-encoders and restricted Bolzmann machines (RBMs) Mesnil et al. (in press). A general methodology seems to have emerged. Most top ranking participants used simple normalizations (like variable standardization and/or data sphering using PCA) as a first step, followed by one or several layers of non-linear processing (stacks of auto-encoders, RBMs, KPCA, and/or clustering). Finally, "transduction" played a key role in winning first place: either the whole preprocessing chain was applied directly to the final evaluation data (this is the strategy of Fabio Aiolli who won first place in phase 1 Aiolli, in press); or alternatively, the final evaluation data, preprocessed with a preprocessor trained on development+validation data, was post-processed with PCA (so-called "transductive PCA" used by the LISA team, who won the second phase Mesnil et al., in press).

Acknowledgments

Our generous sponsors and data donors and our dedicated advisors, beta testers listed on our challenge website (http://clopinet.com/ul) are gratefully acknowledged. This project is part of the DARPA Deep Learning program and is an activity of the Causality Workbench supported by the Pascal network of excellence funded by the European Commission and by the US National Science Foundation under Grant N0. ECCS-0725746. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding agencies.

References

Aiolli, F. (2012). Transfer learning by kernel meta-learning. In *Unsupervised and Transfer Learning Workshop, in Conjunction with ICML 2011*. JMLR W&CP (in press).

Bengio, Y. (2009). Learning deep architectures for AI. *Foundations and Trends in Machine Learning*, *2*(1), 1–127. also published as a book. Now Publishers, 2009.

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5–32.

ChaLearn. Challenges in machine learning. [Online]. Available: http://www.chalearn.org/challenges.html.

Guyon, I., et al, (Eds.), (2012). *Unsupervised and Transfer Learning Workshop, in Conjunction with ICML 2011*. JMLR W&CP (in press). http://jmlr.csail.mit.edu/proceedings/papers.

Guyon, I., Dror, G., Lemaire, V., Taylor, G., & David W, A. (2011). Unsupervised and transfer learning challenge. In *The 2011 International Join Conference on Neural Networks*. pp. 793–800.

Lewis, D. D., Yang, Y., Rose, T. G., Li, F., Dietterich, G., & Li, F. (2004). Rcv1: a new benchmark collection for text categorization research. *Journal of Machine Learning Research*, *5*, 361–397.

Liu, C., Xie, J., Xiong, H., & Ge, Y. (2012). Stochastic unsupervised learning on unlabeled data. In *Unsupervised and Transfer Learning Workshop, in conjunction with ICML 2011*. JMLR W&CP (in press).

Mesnil, G. et al. (2012). Unsupervised and transfer learning challenge: a deep learning approach. In *Unsupervised and Transfer Learning Workshop, in conjunction with ICML 2011*. JMLR W&CP (in press).

Moghaddam, R. F., Cheriet, M., Adankon, M. M., Filonenko, K., & Wisnovsky, R. (2010). IBN SINA: a database for research on processing and understanding of Arabic manuscripts images. *Document Analysis System*, 11–18.

Pan, S. J., & Yang, Q. (2010). A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, *22*(10), 1345–1359.

Raina, R., Battle, A., Lee, H., Packer, B., & Ng, A.Y. (2007). Self-taught learning: transfer learning from unlabeled data. In *Proceedings of the Twenty-Fourth International Conference on Machine Learning*.