Deep-Learning Domain Adaptation Techniques for Credit Cards Fraud Detection

Chapter · January 2019

DOI: 10.1007/978-3-030-16841-4_8

Deep-Learning Domain Adaptation Techniques for Credit Cards Fraud Detection

Bertrand Lebichot[1](✉), Yann-Aël Le Borgne[1], Liyun He-Guelton[2], Frédéric Oblé[2], and Gianluca Bontempi[1]

[1] Machine Learning Group, Computer Science Department, Faculty of Sciences, ULB, Université Libre de Bruxelles, City of Brussels, Belgium
bertrand.lebichot@ulb.ac.be

[2] High Processing & Volume, R&D, Worldline, Lyon, France
http://mlg.ulb.ac.be

Abstract. Although the incidence of credit card fraud is limited to a small percentage of transactions, the related financial losses may be huge. This demands the design of automatic Fraud Detection Systems (FDS) able to detect fraudulent transactions with high precision and deal with the heterogeneous nature of the fraudster behavior. Indeed, the nature of the fraud behavior may strongly differ according to the payment system (e.g. e-commerce or shop terminal), the country and the population segment. Given the high cost of designing data-driven FDSs, it is more and more important for transactional companies to reuse existing pipelines and adapt them to different domains and contexts: this boils down to a well-known problem of transfer learning.

This paper deals with deep transfer learning approaches for credit card fraud detection and focuses on transferring classification models learned on a specific category of transactions (e-commerce) to another (face-to-face). In particular we present and discuss two domain adaptation techniques in a deep neural network setting: the first one is an original domain adaptation strategy relying on the creation of additional features to transfer properties of the source domain to the target one and the second is an extension of a recent work of Ganin *et al.*

The two methods are assessed, together with three state-of-the-art benchmarks, on a five-months dataset (more than 80 million e-commerce and face-to face transactions) provided by a major card issuer.

Keywords: Fraud detection · Domain adaptation · Transfer learning

This work was supported by the DefeatFraud project funded by Innoviris (2017-R-49a). We thank this institution for giving us the opportunity to conduct both fundamental and applied research. B. Lebichot also thanks LouRIM, Université catholique de Louvain, Belgium for their support.

L. Oneto et al. (Eds.): INNSBDDL 2019, INNS 1, pp. 78–88, 2020.
https://doi.org/10.1007/978-3-030-16841-4_8

1 Introduction

Global card fraud losses amounted to 22.8 billion US dollar in 2017 and is foreseen to continue to grow [20]. In recent years, machine learning techniques appeared as an essential component of any detection approach dealing automatically with massive amounts of transactions [12]. The existing work showed however that a detection strategy needs to take into account some peculiarities of the fraud phenomenon [10,11]: unbalancedness (frauds are less than 1% of all transactions), concept drift (typically due to seasonal aspects and fraudster strategies) and the big data and streaming nature [4]. Disregarding those aspects might lead to high false alert rate, low detection accuracy or slow detection (see [1] for more details). As a result the design of an accurate Fraud Detection System (FDS) goes beyond the integration of some conventional off-the-shelf learning libraries and requires a deep understanding of the fraud phenomenon. This means that the reuse of existing FDS in new settings, like a new market (e.g. with a different fraud ratio) or a new payment system, is neither immediate nor straightforward.

This paper depicts transfer learning strategies [23] in the adaptation of existing and effective models to new domains. In particular we focus on the heterogeneous nature of the credit-card transactions, related to the physical presence of the card holder, which distinguishes between face-to-face (F2F) and e-commerce (EC) settings. Face-to-face transactions occur when the buyer and merchant have to physically meet in order to complete a purchase. In e-commerce (e.g. exchange of goods or services through a computer network, like internet) transactions can take place when the card holder is not physically with the merchant.

E-commerce fraud detection settings have been more studied in literature [3,12,15,22,29] than face-to-face ones [21]. Though F2F frauds are typically less frequent because personal identification number (PIN) is often required, their impact is not negligible and worthy of consideration. Well-known examples of F2F frauds are due to *skimming* i.e. the criminal action retrieving the card holder information from the card magnetic strip.

The differences between the EC and F2F setting, notably in terms of different ratios of genuine vs. fraudulent transactions and different attitudes of the fraudsters, are reflected in different statistical properties of the two detection tasks. At the same time much of the detection process is similar, for instance in terms of feature representation of the transactions. It is therefore important for card issuer companies to understand how much of the modelling and design effort done in setting up an accurate detection system for EC (source domain) can be reused and transferred to F2F (target domain).

We first review the topic of transfer learning and presents some state-of-the art *domain adaptation* methods which can be used to transfer the knowledge learned during EC fraud detection to enhance the detection of F2F frauds. Then, it presents two original contributions: (i) the proposal of an original transfer strategy relying on the creation of ad-hoc features (related to the marginal and the conditional distribution) to improve the transfer from the source to the target domain, (ii) the customization of an existing adversarial deep-learning strategy (typically used in image recognition task) to the specific task of credit card fraud detection (e.g. taking into account issues of unbalancedness and concept drift).

To the best of our knowledge this is the first paper assessing the quality of transfer learning techniques for credit card fraud detection. Another specificity of the paper is the extensive assessment procedure carried out on a five-months real-life dataset obtained from the major credit card issuer in Belgium.

The rest of this paper is structured as follows: Sect. 2 introduces background and notation. Section 3 reviews related work. Section 4 details the methodological contributions of the paper. Experimental comparisons are presented and analyzed in Sect. 5 and Sect. 6 discusses the results. Section 7 concludes this paper.

2 Background and Notation

Transfer learning (TL) is a crucial aspect of real-world learning: for instance, learning to recognize apples might help to recognize pears, or knowing to play piano might help learning electric organ [23]. Suppose also that you trained a learning machine to label a website on the basis of existing websites. Transfer learning could help to adapt the learner to deal with brand new websites [25]. The rationale of transfer learning is to fill the gap between two learning tasks and to reuse as much as possible what was learned from the first task to better solve the second one. More precisely, given a source domain D_s and learning task T_s, a 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 knowledge in D_s and T_s where $D_s \neq D_t$ or $T_s \neq T_t$. Transfer learning requires at least a change in domains or in tasks [23,26]:

- A *domain* D is defined as a tuple $(\mathbf{X}, P_{\mathbf{X}}(\cdot))$, where $\mathbf{X}$ denotes the multivariate input and $P_{\mathbf{X}}(\cdot)$ its marginal probability distribution.
- Given a specific domain, a *task* T is defined as a tuple $(\mathbf{Y}, f(\cdot))$ where $\mathbf{Y}$ is the label space and $f(\cdot)$ summarizes the conditional dependency (either the regression function or conditional distribution).

Domain adaptation (DA) is a sub-field of transfer learning: there is a change in the domain $D_s \neq D_t$ but the task remains the same $T_s = T_t$. In this paper, the task is the same, fraud detection, but there is a change of domain. In this context, by *source* (with subscript s) we denote the original domain (e-commerce) while *target* (with subscript t) refers to the new domain (face-to-face).

The transaction dataset is a collection of n vectors $\mathbf{x}_i$ of size m where m is the number of features (or attributes). The features are identical in the source and the target domain but the marginal distribution changes between them. Finally we define $\mathbf{y}$ as the column vector containing the class labels (fraudulent or genuine) of the n transactions.

3 Related Work

Transfer learning and domain adaptation have been widely studied in the last 20 years but none, to our knowledge, were devoted to FDS. There are four main classes of DA techniques in literature [23–25] which are described below.

Instance weighting for covariate shift: the *covariate shift* scenario refers to a non-stationary environment, that is a setting where the input distribution changes while the conditional distribution remains the same [27]). This scenario may occur when the training data has been biased toward one region of the input space or is selected in a non-I.I.D. manner. Instance weighting methods aim to weight samples in the source domain to match the target domain. For example the paper [23] proposes to estimate weights of the source domain, trying to make the weighted distributions of both domains as similar as possible.

Self-labeling methods: they include unlabeled target domain samples in the training process, initialize their labels and then iteratively refine them. This is often done using Expectation-Maximization (EM) algorithms (for example TrAdaBoost [9]). Hard versions add samples with specific labels while others [28] assign label confidences when fitting the model. While efficient, the EM procedure can be really computationally intensive, especially with large datasets.

Feature representation methods: they aim to find a new feature representation for the data and belong to two categories. Distribution similarity approaches aim explicitly to make the source and target domain sample distributions similar, either by penalizing or removing features whose statistics vary between domains or by learning a feature space projection in which a distribution divergence statistic is minimized [16,17]. On the other hand, latent feature approaches aim to construct new features using (often unlabeled) source and target domain data or, more widely, to find an abstracted representation for domain-specific features [13,17]. We discuss an easy way [13] to do so in Sect. 4.

Cluster-based learning methods: these methods construct a graph where the labeled and unlabeled samples are nodes. Edge weights, between samples, are based on their similarity. Labels are then propagated according to the graphs (e.g. by means of graph-based classification). The main assumption is that samples connected by high-density paths are likely to have the same label if there is a high density path between them [17]. Also those methods may be highly computationally intensive, especially when working with large graphs.

Furthermore, deep neural networks methods (DNN) have been widely used for TL and DA: their multi-layer nature can capture the intricate non-linear representations of data, and provide useful level features for transfer learning [23]. Multitask learning [2] can be easily implemented by DNN, by training two or more related tasks with a network sharing inputs and hidden layers but having separate output layers. As far as domain adaptation is concerned, hidden layers trained by the source task can be reused on a different target task. For the target task model, only the last classification layer needs to be retrained, though any layer of the new model could be fine-tuned if needed [23]. In other configurations, the hidden parameters related to the source task can be used to initialize the target model [8]. Autoencoders can also be used to gradually changing the training distribution: In [7], a sequence of deep autoencoders are trained layer-by-layer, while gradually replacing source-domain samples with target-domain samples. In [18] the authors simply trains a single deep autoencoder for both domains. Finally, Ganin and al. used DNN in an adversarial way (we will discuss more extensively this approach in Sect. 4) to tackle domain adaptation [16].

Table 1. This table summarizes the five considered methods of this paper. More details about the strategy and parameters can be found in Sect. 4.

Acronym	Strategy	Train set	Test set	Selected parameter
BDNN	Baseline	F2F	F2F	-
NDNN	Naive	EC	F2F	-
FEDADNN	Imputation	EC + F2F	F2F	-
AugDNN	Add EC-related features	Extended F2F	Extended F2F	$\lambda = 10$
AdvDNN	Adversarial	EC + F2F	F2F	$n_{PC} = 2$

4 Transfer Learning Strategies for Fraud Detection

As discussed in the previous section, several approaches may be taken into account to transfer information related to a source task to a target one. Though in a realistic situation none or very few labeled training samples could be available in the target domain, for the sake of assessment, we consider here an experimental setting where training samples are available for both the source and the target tasks. The target dataset is splitted in a training and test portion. The test portion makes possible a sound paired assessment of all the considered strategies while the training portion enables us to assess how much improvement may derive from integrating the source dataset with (part of) the target dataset.

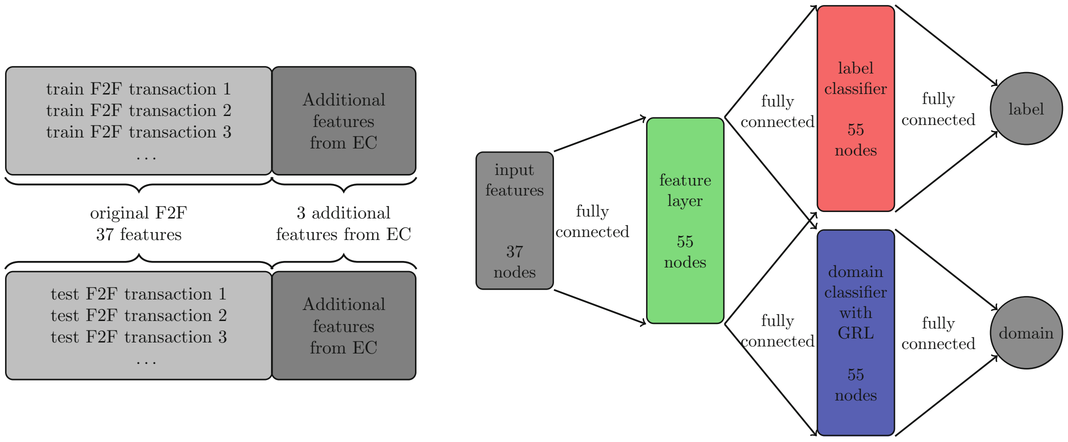
Table 1 lists the alternative strategies which differentiate in terms of training set (different combinations of source and target) and transfer methodology. To avoid biases related to the learning machine, all strategies share the same DNN topology composed of two fully connected hidden layers and implemented on Keras [6]. Based of preliminary results (not reported here), we set the number of neurons in the hidden layers n_h to 1.5 times the number of features (here, 37).

The five considered methods are (code on https://github.com/B-Lebichot):

- *BDNN:* this is our baseline DNN classifier ($n_h = 55$). The train and test data are composed of target (F2F) samples: no transfer learning in this baseline.
- *NDNN:* this is the naive strategy which simply consists in training the same DNN ($n_h = 55$) as BDNN on the source (EC) dataset and test it on the target (F2F) testset. This approach is also often considered in literature as a baseline [13] to assess the added value of a transfer learning strategy.

- *FEDADNN:* this is a basic feature representation method (Sect. 3) which uses three versions of the original feature set: a general version, a source-specific version and a target-specific version [13].
  Each source column-feature $\mathbf{x_s}$ is simply replaced by $\phi^s(\mathbf{x_s}) = \langle \mathbf{x_s}, \mathbf{x_s}, \mathbf{0} \rangle$ and each target column-feature $\mathbf{x_t}$ is replaced by $\phi^t(\mathbf{x_t}) = \langle \mathbf{x_t}, \mathbf{0}, \mathbf{x_t} \rangle$. Where $\mathbf{0}$ is a vector full of zeros. Where $\phi^s(\mathbf{x_s})$ is the source mapping and $\phi^s(\mathbf{x_t})$ is the target mapping (roughly this is $\langle$general, source-specific, target-specific$\rangle$).
  This strategy allows to express both domains in an extended feature space, imputing missing values. The augmented source data therefore contains only general and source-specific versions while the augmented target data contains both general and target-specific versions. Finally, ϕ^t is used to obtain the test set from the original target data. As the number of features tripled compared to BDNN/NDNN, we set $n_h = 165$.
- *AugDNN:* this is an original technique whose rationale is to use information from the source domain (e.g. conditional distribution, marginal input distribution) as additional features of the classifier (for both training set and test set). This strategy allows the classifier to learn from data how the relatedness [5] between source and target samples is associated to the classification output. The main difficulty is that such information is not explicitly available but can only be estimated. This is the reason why we fit both a classifier and a dimensionality reduction to extract from the source domain information about the conditional distribution and the input distribution (see Fig. 1(a) for an illustration). Other variants were studied (Gaussian mixture models, ...) but we select the more informative subset of features. As a result we add three new features (denoted *Aug1* to *Aug3*) to the original feature set (therefore $n_h = 60$) obtained as follows:
  - We train a regular DNN source classifier. The first feature (*Aug1*) is then the predicted activation value (i.e. estimated conditional probability) of the output neuron for each F2F sample. [13] used a similar idea but used the binary predicted value instead.
  - We build a principal component analysis (e.g. PCA [19]) on the source training set. The projections of the F2F transactions on the first two PCs ($n_{PC} = 2$) return *Aug2 and Aug3.*

  The aim of adding such features is to encode in the training set the relatedness between the source and target distributions, both from a marginal and conditionally dependent perspective.
- *AdvDNN:* this is an adaptation of Ganin *et al.* [16] approach to the fraud detection setting. The rationale is that the prediction model must use features that cannot discriminate between the source and target domains. The original approach was used for image recognition and combines a labeled source domain and unlabeled target domain while here both source and target are labeled. The method learns domain invariant features by jointly optimizing the feature layer from the label predictor (here genuine versus fraudulent) and the domain label (here F2F versus EC) predictor. The domain classifier



(a) *AugDNN* (b) *AdvDNN*

Fig. 1. Illustration of methods *AugDNN* and *AdvDNN*. For (b), $n_h = 55$ and all successive layers are fully connected. Notice that removing the domain classifier reduces the network to the BDNN and NDNN baselines.

uses a gradient reversal layer (GRL) and a few fully connected layers [16]. The effect of the GRL is to multiply all domain-related gradients by a negative constant λ during back-propagation.

During the training, the feature layer is optimized to both minimize the label classifier loss and to maximize (due to GRL) the domain classifier loss. This approach promotes the emergence of features that are discriminative for the main learning task on the source domain and non-discriminative with respect to the domain tag [16]. A network illustration can be found on Fig. 1(b).

5 Experimental Comparisons

In this section, the different methods of Table 1 are compared on a real-life credit card transaction dataset obtained from our industrial partner.

The source database is composed of 37,882,367 e-commerce transactions (138 days: 85 training, 3 validation days and 50 test days) and 37 features. Validation days are used to tune Λ. The fraud ratio is 0.366%. The target database is composed of 47,619,852 face-to-face transactions (the same days and features as the source database). The fraud ratio is 0.033%.

The accuracy indicator is Precision@100 (Pr@100) (for details on such measure see [12]) which reports the number of true compromised cards among 100 investigated. The number 100 is chosen since this is compliant with the daily effort of the team of human investigators who manually check the transactions. For a deeper discussion on accuracy indicators for FDS, see [4,10,12,22]. Transaction-based precision is sometimes used instead of card-based precision: We obtained similar conclusions using transaction-based precision.

6 Discussion

Figures 2 and 3 compare methods described on Table 1 through a Friedman/Nemenyi test [14] for cards-based Pr@100. We adopt a sliding window approach (see Subsect. 5). Friedman null hypothesis is rejected with $\alpha = 0.05$ and Nemenyi critical difference is equal to 0.837. A method is considered as significantly better than another if its mean rank is larger by more than this amount.

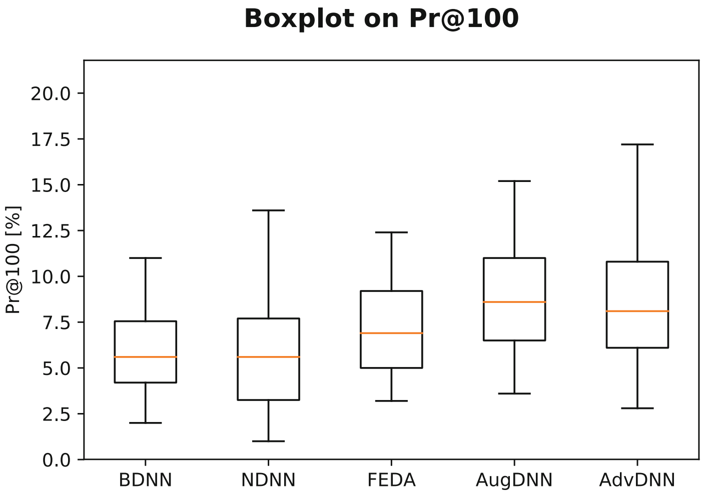


Fig. 2. Boxplots representing the card-based precision@100 for each method (50 days with one precision score per day). Notice that the a priori fraud probability is only 0.033%. The largest relative mean increase (NDNN versus AugDNN) is +59%.

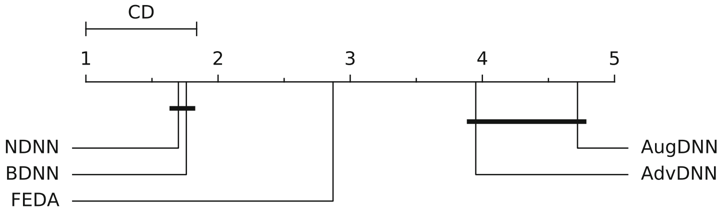


Fig. 3. Mean rank (the higher, the better) and critical difference (CD) of the Friedman/Nemenyi test for the five methods. A method is considered as significantly better than another if its mean rank is larger by more than the critical difference $CD = 0.838$.

From Fig. 2, it appears that the F2F detection accuracy is much lower than the EC one. In previous works [4,12] we showed that the EC Pr@100 can attain 50% when no domain adaptation is involved. This is explained by the fact that the F2F a priori fraud probability is ten times lower than for EC.

We also observe that BDNN and NDNN (the two baselines) are not such different, showing that both tasks are actually related (this is confirmed by the Friedman/Nemenyi test on Fig. 3). However, the three considered domain adaptation methods increase the accuracy with respect to the baselines.

From Fig. 3, the FEDADNN algorithm is clearly less accurate than the AugDNN and AdvDNN approach. Those two last methods cannot be significantly discriminated on our data by the Friedman/Nemenyi test. However, a Wilcoxon test between AugDNN and AdvDNN indicates that AugDNN significantly outperforms with $\alpha < 0.05$. Overall, AugDNN could be considered as the best method emerging from our experimental assessment.

All experiments were carried on a server with 10 cores, 256 GB RAM and three Asus GTX 1080 TI. These are the related execution times (feature manipulation and classification only): BDNN, NDNN, FEDADNN, AugDNN, and AdvDNN runs in 12, 16, 23, 41, and 46 min, respectively. Note that accuracy comes at the price of increase execution time and that AugDNN is slightly superior to AdvDNN both in terms of accuracy and run time.

7 Conclusion

The paper studied the use of domain adaptation strategies in transaction-based fraud detection systems. In particular, we considered e-commerce transactions as the source domain and face-to-face transactions as the target domain.

We introduced two original methods. The first adds e-commerce related features (both predictive and distribution-related) to the face-to-face transactions to improve predictions. The second is an adaptation of the work from [16]. This method learns domain invariant features by jointly optimizing the underlying feature layer from the fraud tag predictor (here genuine versus fraudulent) and the domain tag (face-to-face versus e-commerce).

Those two methods, and three others, were tested on a five-months (more than 80 millions of transactions) real-life e-commerce and face-to-face credit card transaction dataset obtained from a large credit card issuer in Belgium. The second proposed method outperforms all the considered approaches and the first proposed method comes second (in terms of performance and run time). Those results are shown to be significant using statistical tests.

Future work will focus on scenarios characterized by a low ratio of labeled transactions in the target domain. Better density estimation (using for example Gaussians mixture models) for the source distribution of our first method (AugDNN) will also be studied.

References

1. Abdallah, A., Maarof, M.A., Zainal, A.: Fraud detection system. J. Netw. Comput. Appl. **68**, 90–113 (2016)
2. Ahmed, A., Yu, K., Xu, W., Gong, Y., Xing, E.: Training hierarchical feed-forward visual recognition models using transfer learning from pseudo-tasks. In: ECCV (3), pp. 69–82 (2008)
3. Bolton, R., Hand, D.: Statistical fraud detection: a review. Stat. Sci. **17**, 235–249 (2002)

4. Carcillo, F., Dal Pozzolo, A., Le Borgne, Y.A., Caelen, O., Mazzer, Y., Bontempi, G.: SCARFF: a scalable framework for streaming credit card fraud detection with spark. Inf. Fusion **41**(C), 182–194 (2018)
5. Caruana, R.: Multitask learning. Mach. Learn. **28**(1), 41–75 (1997)
6. Chollet, F., et al.: Keras (2015). https://keras.io
7. Chopra, S., Balakrishnan, S., Gopalan, R.: DLID: deep learning for domain adaptation by interpolating between domains. In: ICML Workshop on Challenges in Representation Learning (2013)
8. Ciresan, D.C., Meier, U., Schmidhuber, J.: Transfer learning for Latin and Chinese characters with deep neural networks. In: IJCNN, pp. 1–6. IEEE (2012)
9. Dai, W., Yang, Q., Xue, G.R., Yu, Y.: Boosting for transfer learning. In: Proceedings of the 24th International Conference on Machine Learning, ICML 2007, pp. 193–200. ACM (2007)
10. Dal Pozzolo, A., Boracchi, G., Caelen, O., Alippi, C., Bontempi, G.: Credit card fraud detection and concept-drift adaptation with delayed supervised information. In: Proceedings of the International Joint Conference on Neural Networks, pp. 1–8. IEEE (2015)
11. Dal Pozzolo, A., Boracchi, G., Caelen, O., Alippi, C., Bontempi, G.: Credit card fraud detection: a realistic modeling and a novel learning strategy. IEEE Trans. Neural Netw. Learn. Syst. **29**(8), 3784–3797 (2018)
12. Dal Pozzolo, A., Caelen, O., Le Borgne, Y.A., Waterschoot, S., Bontempi, G.: Learned lessons in credit card fraud detection from a practitioner perspective. Expert. Syst. Appl. **10**(41), 4915–4928 (2014)
13. Daume III, H.: Frustratingly easy domain adaptation. In: Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, pp. 256–263. Association for Computational Linguistics, Prague, Czech Republic, June 2007
14. Demsar, J.: Statistical comparaison of classifiers over multiple data sets. J. Mach. Learn. Res. **7**, 1–30 (2006)
15. Fawcett, T., Provost, F.: Adaptive fraud detection. Data Min. Knowl. Discov. **1**, 291–316 (1997)
16. Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., Lempitsky, V.: Domain-adversarial training of neural networks. J. Mach. Learn. Res. **17**(1), 2096–2030 (2016)
17. Gao, J., Fan, W., Jiang, J., Han, J.: Knowledge transfer via multiple model local structure mapping. In: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2008, pp. 283–291. ACM, New York (2008)
18. Glorot, X., Bordes, A., Bengio, Y.: Domain adaptation for large-scale sentiment classification: a deep learning approach. In: Proceedings of the Twenty-eight International Conference on Machine Learning, ICML (2011)
19. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning, 2nd edn. Springer, New York (2009)
20. HSN Consultants, Inc.: The Nilson report (consulted on 2018-10-23) (2017). https://nilsonreport.com/upload/content_promo/The_Nilson_Report_Issue_1118.pdf
21. Jurgovsky, J., Granitzer, M., Ziegler, K., Calabretto, S., Portier, P.E., He, L., Caelen, O.: Sequence classification for credit-card fraud detection. Expert. Syst. Appl. **100**, 234–245 (2018)
22. Lebichot, B., Braun, F., Caelen, O., Saerens, M.: A graph-based, semi-supervised, credit card fraud detection system, pp. 721–733. Springer, Cham (2017)

23. Lu, J., Behbood, V., Hao, P., Zuo, H., Xue, S., Zhang, G.: Transfer learning using computational intelligence: a survey. Knowl. Based Syst. **80**, 14–23 (2015)
24. Margolis, A.: A literature review of domain adaptation with unlabeled data. Technical report, University of Washington (2011)
25. Pan, S.J., Yang, Q.: A survey on transfer learning. IEEE Trans. Knowl. Data Eng. **22**(10), 1345–1359 (2010)
26. Pan, S.J., Yang, Q., et al.: A survey on transfer learning. IEEE Trans. Knowl. Data Eng. **22**(10), 1345–1359 (2010)
27. Sugiyama, M., Kawanabe, M.: Machine Learning in Non-Stationary Environments: Introduction to Covariate Shift Adaptation. The MIT Press, Cambridge (2012)
28. Tan, S., Cheng, X., Wang, Y., Xu, H.: Adapting Naive Bayes to domain adaptation for sentiment analysis. In: Proceedings of the 31th European Conference on IR Research on Advances in Information Retrieval, ICML 2009, pp. 337–349. Springer (2009)
29. Van Vlasselaer, V., Bravo, C., Caelen, O., Eliassi-Rad, T., Akoglu, L., Snoeck, M., Baesens, B.: APATE: a novel approach for automated credit card transaction fraud detection using network-based extensions. Decis. Support Syst. **75**, 38–48 (2015)