Domain Adversarial Training for Aspect-Based Sentiment Analysis

Joris Knoester[1], Flavius Frasincar[1] , and Maria Mihaela Truşcă[2(✉)]

[1] Erasmus University Rotterdam, Burgemeester Oudlaan 50, 3062 PA Rotterdam,
The Netherlands
`frasincar@ese.eur.nl`
[2] Bucharest University of Economic Studies, 010374 Bucharest, Romania
`maria.trusca@csie.ase.ro`

Abstract. The continuously expanding digital possibilities, increasing number of social media platforms, and growing interest of companies in online marketing increase the importance of Aspect-Based Sentiment Analysis (ABSA). ABSA focuses on predicting the sentiment of an aspect in a text. In an ideal scenario, we would have labeled data for every existing domain, but acquiring annotated training data is costly. Transfer learning resolves this issue by building models that can be employed in different domains. The proposed work extends the state-of-the-art LCR-Rot-hop++ model for ABSA with the methodology of Domain Adversarial Training (DAT) in order to create a deep learning adaptable cross-domain structure, called the DAT-LCR-Rot-hop++. The major advantage of the DAT-LCR-Rot-hop++ is the fact that it does not require any labeled target domain data. The results are obtained for six different domain combinations with testing accuracies ranging from 37% up until 77%, showing both the limitations and benefits of this approach. Once DAT is able to find the similarities between domains, it produces good results, but if the domains are too distant, it is not capable of generating domain-invariant features.

Keywords: Aspect-Based Sentiment Classification · Transfer learning · Adversarial Training

1 Introduction

Ever since the introduction of the first social media platform, Six Degrees in 1997 [10], people have communicated via digital platforms. By last year, over 3.7 billion people, which accounts for 48% of the world population, have exchanged information on social media [29]. Due to the continuously growing society of Web users, more companies are becoming aware of the essential role that social media plays in brand image. The increase in users has expanded the amount of opinionated messages, which are valuable if correctly analysed.

Aspect-Based Sentiment Analysis (ABSA) classifies a person's feeling towards specific aspects [30]. ABSA is concerned with target extraction (TE),

aspect detection (AD), and target sentiment classification (SC) [26]. This paper concentrates on the last task, SC, classifying the user's sentiment with respect to the aspects, and is called Aspect-Based Sentiment Classification (ABSC) [2]. As an example, we provide the sentence "the atmosphere and service were terrible, but the food was good". In this case, the "atmosphere", "service", and "food" are all *target words* and "terrible" and "good" are the expressions that give *context* to these target words to be able to label the targets as positive, neutral, or negative. As one can notice, the context around the target words is essential to capture the explicit aspect sentiments.

There are multiple practical applications of ABSC. By evaluating and deciding which features need improvement, a company can apply specific enhancements to their products, efficiently improving their customer services. At the same time, social media platforms such as Facebook and Twitter can implement ABSC on tweets and messages, and sell this valuable information to the marketing department of multinationals. In addition, financial firms can apply ABSC to forecast the feelings of financial individuals towards the economic market and thereby predict future stock movements. This would have been extremely beneficial for Melvin Capital before the whole GameStop phenomena at Reddit and might have prevented enormous losses [3]. Last, knowing the opinion of previous customers can help potential clients make better-informed buying decisions.

An issue that has gained much attention recently is the limited availability of labeled data. Generating new labeled data for specific domains is expensive, time-consuming, and requires manual labour. In order to decrease the dependence on labeled data, transfer learning, also called cross-domain learning, is a valuable solution [20]. This approach concentrates on training a model on a related source domain and then predicting for a different target domain. Several state-of-the-art cross-domain models rely on Domain-Adversarial Neural Networks (DANN) introduced in [8]. These neural networks are applied on diverse tasks, ranging from textual entailment analysis [14] to image classification [39]. However, little research is available on ABSA using Domain-Adversarial Training (DAT). To our knowledge, there is limited research on training domain-invariant features for Aspect-Based Sentiment Classification (ABSC) using DANN. Prior methods for sentiment classification apply a Support Vector Machine (SVM) [21] in order to predict the sentiments, but due to its shortcomings, this system was replaced by knowledge-based models [28] and deep learning algorithms [15]. Whereas the deep learning methods are flexible, knowledge-based models require more manual labour, but achieve better results [18]. Since a combination of both approaches benefits from the advantages of both solutions, several researchers merge the methods into a hybrid model [31,34].

The proposed methodology of this paper is based on the hybrid HAABSA++ model proposed in [32], a state-of-the-art ABSC approach that produces excellent results for the commonly used SemEval 2015 [23] and 2016 [24] data sets, attaining a classification accuracy of 81.7% and 87.0%, respectively. Nevertheless, our study focuses on the development of a neural network that can classify texts on multiple different domains. For this reason, the knowledge-based ontology

part of the HAABSA++ approach is eliminated, leaving the LCR-Rot-hop++ model based on [42]. This model is expanded by applying the structure of DAT as proposed by [8]. More specifically, the neural network is trained concurrently on labeled instances of a source domain and unlabeled instances from a target domain. We call the newly established system Domain Adversarial Training LCR-Rot-hop++, abbreviated as DAT-LCR-Rot-hop++. All source code and data can be retrieved from https://github.com/jorisknoester/DAT-LCR-Rot-hop-PLUS-PLUS.

The main contribution of this research on the current literature is the capability of predicting aspect-based sentiment classifications on target aspects without requiring annotated target data by employing DANN. Following the general DANN approach, this paper uses both labeled source and unlabeled target domain aspects to obtain domain indiscriminative representations. To the best of our knowledge, there is limited research on extending an advanced neural network as LCR-Rot-hop++ for ABSC with DANN in a cross-domain setting.

The rest of this paper is structured as follows. First, Sect. 2 gives the relevant literature concerned with ABSC, the different components of the LCR-Rot-hop++ model, and the ideas behind transfer learning. Second, Sect. 3 provides a short description of the data together with a couple of descriptive statistics. Third, Sect. 4 concentrates on the methodology of this research for which the results are reported in Sect. 5. Last, Sect. 6 provides our conclusion and suggestions for future work.

2   Related Work

ABSC is concerned with classifying a person's sentiment towards specific aspects in a sentence. One of the first works on sentiment analysis is presented in [22] and sentiment analysis has been a hot topic ever since. While the traditional methods were mainly developed based on knowledge-based systems providing higher prediction scores for domain-specific documents [18], the more recent ones tend to rely more on deep learning solutions. Despite being considered alternative solutions, in [36] it is observed that the two approaches are in fact complementary. Consecutively, a hybrid method which incorporates both the domain ontology and a deep learning neural network is introduced in [27]. After several additional improvements, this has resulted in the HAABSA++ method [32].

HAABSA++ method has two steps so that if the ontology is unable to provide reliable results, the LCR-Rot-hop++ neural network is used as a backup. As the topic of our work is transfer learning, we focus only on the neural network. LCR-Rot-hop++ is actually a bi-directional Long Short-Term Memory (Bi-LSTM) model accompanied by an attention. This attention layer is able to put focus on specific parts of a sentence, thereby limiting the influence of the less important words. On top of this attention layer, a hierarchical attention layer is employed, enabling the model to process the text on sentence-level, bringing together the local sentence representations. The last component of the LCR-Rot-hop++ model is its rotary system that increases the interaction between targets

and contexts by sharing information in order to capture the most indicative sentiment words.

Transfer learning [20] is a machine learning technique that focuses on storing information from one data set and applying this knowledge on another. Because obtaining annotated data is costly and time-consuming, new models must be developed to provide reliable results for multiple domains. The variety of methods of transfer learning is continuously expanding. One of the proposed solutions focuses on freezing the first layers of an LSTM neural network [4]. This approach is based on the fact that the source domain contains valuable universal information and higher layer neurons tend to specialize more towards the target domain, while the lower hidden layers generate more common word features [37]. A state-of-the-art method, BertMasker [38] uses the idea of masking [6] for their BERT Base network. BertMasker is able to mask domain-related words. This transforms the remaining sentence text to be domain-invariant, but at the same time still maintains its most sentiment-explicit words. Another solution scientists developed is a domain adapting network by creating counterfactual features [13]. These counterfactual depictions reduce the inductive bias of the source domain. The designed positive (negative) counterfactuals bridge the dimensional gap between the positive (negative) classified instances of the source and target domain. Different from the previous works, in this research we apply the methodology of Generative Adversarial Networks (GAN). This last solution is introduced in [9] and has shown superior performance in a broad range of scientific areas, such as image classification [5], event detection [12], and textual cross-domain sentiment classification [41]. Applying the logic of GAN, DANN is introduced in [7]. A DANN model is able to perform machine learning tasks on unlabeled target domain data, while trained on a labeled source domain with a relatively similar distribution, both in terms of polarity distribution and batch size [7]. The advantage that one does not need annotated target data makes DANN very valuable for future cross-domain deep learning problems and is therefore an important contribution to the existing machine learning techniques.

3   Data

In this paper, two different data sets are used. These are the Semantic Evaluation (SemEval) 2014 [25], and the Amazon/LibraryThing (ALT) 2019 [17]. SemEval 2014 includes information about the restaurant and laptop domain, and ALT contains the data for the book domain. The reviews of the data sets are divided into single sentences, consisting of one or more aspects and sentimental context words, which are used to classify the polarities (positive, neutral, or negative). The partitioning of the data into a training and test set is done as follows. The aspects are divided into 80% training and 20% testing. The training set consists of 80% pure training and 20% validation to compute the optimal values for the hyperparameters. The results of the split into training and test data are presented in Table 1.

Table 1. The distribution of the aspect sentiment polarities of the three domains.

	Restaurant		Laptop		Book	
	Train	Test	Train	Test	Train	Test
Positive	60.1%	65.1%	42.7%	52.4%	25.8%	32.3%
Neutral	17.7%	17.5%	19.8%	26.2%	63.1%	57.1%
Negative	22.2%	17.4%	37.5%	21.4%	11.1%	10.6%
Total	3600	1122	2250	701	2700	804

In terms of data pre-processing, the same approach is applied as introduced in [35] and [42]. The implicitly opinionated review sentences contain a sentiment, but the aspect term is missing. This makes it impossible to perform ABSC using the LCR-Rot-hop++ model. In addition to this, it could occur that an aspect has conflicting sentiments. This happens when there is both negative and positive context towards an aspect. Both the conflicting sentiment and the implicitly opinionated sentences are removed from the datasets.

DAT-LCR-Rot-hop++ requires aspects from two domains to go through the model. These domains are defined as the source domain and the target domain. Only the source domain instances have a sentiment label attached to them. As a result, during training, the aspects of the source domain consist of two labels, the domain class, d, and, the sentiment category, y, while the instances of the target domain only contain a domain class. For testing, the polarity labels of the target domain aspects are determined in order to evaluate the performance of DAT-LCR-Rot-hop++. Our proposed model is concurrently trained on one source and one target domain to obtain domain-invariant features. Results are presented for six different domain combinations, being restaurant-laptop, restaurant-book, laptop-restaurant, laptop-book, book-restaurant, and book-laptop. As an example, the restaurant-laptop model means that the restaurant data set is the source domain and the laptop data set is the target domain. When training is finished, the test instances of the target domain with sentiment labels are fed into the model for evaluation. The performance of DAT-LCR-Rot-hop++ is analysed according to its predicting sentiment accuracy of the target test aspects.

4 Framework

A GAN model [9] generally consists of two additional elements on top of the neural feature extractor, which is the LCR-Rot-hop++ model in this paper. The feature extractor transforms an input sentence to a vector representation that is ought to capture the important characteristics of the sentence. The other two extra elements are the generator and discriminator. A visual representation of our neural network is shown in Fig. 1.

The DAT method introduced in [7] is able to adapt to target domains without any labeled target data. This is done by generating deep features that are
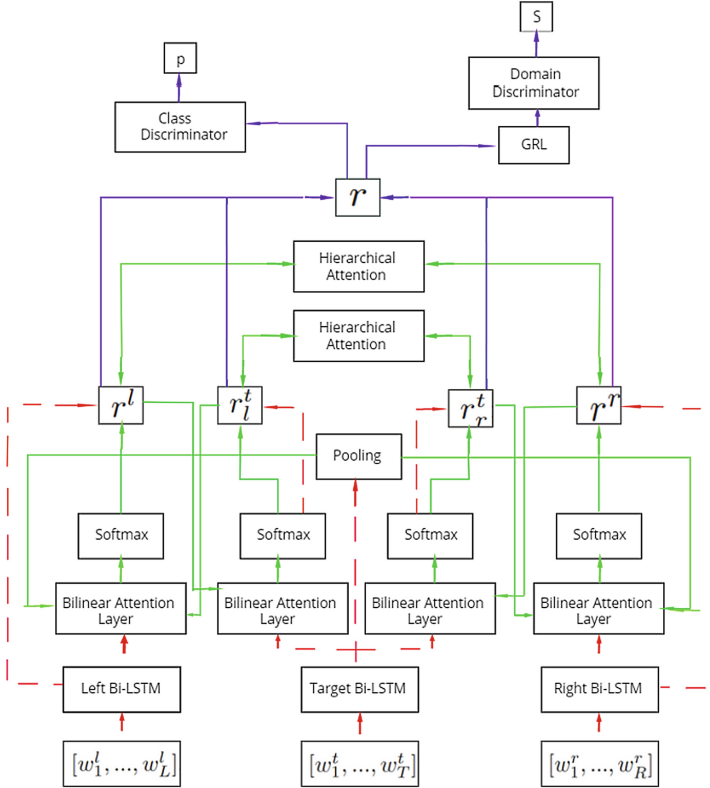
Fig. 1. A visualisation of the LCR-Rot-hop++ model. (Color figure online)

discriminative for the main learning classifying task by using the labeled senti-
ments of the source domain. This method ensures that these representations are
invariant to shifts between the source and target domain in order to be domain
in-discriminative by using the domain class of both the source and target domain.
The proposed DANN solution is revisited in [8], which provides a more detailed
and elaborate description of the mathematics behind the system.

The change to the GAN network in order to conform to a DANN model
means replacing the generator with a Gradient Reversal Layer (GRL). GRL
aims to make the task of the domain discriminator as hard as possible, which is
the direct connection to GAN. In DAT-LCR-Rot-hop++, the loss of the domain
discriminator is passed through GRL, which reverses the gradient before back-
propagation into the feature extractor. This causes the hidden layers of LCR-
Rot-hop++ to react by constructing features which will not be recognised as
a certain domain by the domain discriminator. This process continues until at
some point, the word vectors are completely domain-invariant, which causes the
domain discriminator to be unable of distinguishing the source and target domain
in the shared feature representations. The rest of this section is structured as

follows. Section 4.1 describes the structure of our proposed model and Sect. 4.2 presents the training procedure for our model.

4.1 Structure

The main difference to the original LCR-Rot-hop++ model is the removal of the MLP output layer and its replacement by a domain adversarial component. After the context and target representations are concatenated into r, produced by the feature extractor, this vector is passed into two standard feed-forward MLPs, which are the class discriminator and the domain discriminator. First, the domain discriminator aims to correctly classify the domain of r. The predicted domain is given by s. Classifying the domain is a binary problem with $s = 0$ for source and $s = 1$ for target domain labels. Next, the class discriminator uses a softmax function to compute the probabilities of the sentiment of the aspect, resulting in a 1×3 output vector, p. The polarity that has the largest probability will be chosen as the final sentiment. The sigmoid function is used for the domain prediction because it shows good performance for examining binary cases and is applied by multiple researches in domain discriminators [11,40]. The DAT component is visualised by the dark purple solid arrows in Fig. 1.

The objective is reducing the error term of both the domain discriminator, denoted as $L_d(\theta_f, \theta_d)$, and the class discriminator (sentiment discriminator), denoted as $L_c(\theta_f, \theta_c)$. Here, θ represents the parameters of the feature extractor (LCR-Rot-hop++ without original MLP), the domain discriminator, and the class discriminator, defined by the underscores f, d, and c, respectively. Hence, the objection function to optimise is:

$$\min_\theta \ L_{c,d}(\theta_f, \theta_c, \theta_d) = L_d(\theta_f, \theta_d) + L_c(\theta_f, \theta_c). \tag{1}$$

However, as previously described, the GRL tries to fool the domain discriminator. After the domain is predicted and its parameters, θ_d, are updated, the loss is back-propagated into the feature extractor to change the weights accordingly. But this loss first passes through the GRL, which reverses the gradient by multiplying it with $-\lambda$ in order to hinder the performance of the domain discriminator. The reversing of the gradient forces the hidden layers of the LCR-Rot-hop++ to respond by adjusting their weights in the exact opposite way as desired by the domain discriminator, hereby making the task of the domain classifier more difficult. As a result, the features become more domain indiscriminative. This process leads to the following adjusted loss function:

$$\min_\theta \ L'_{c,d}(\theta_f, \theta_c, \theta_d) = -\lambda L_d(\theta_f, \theta_d) + L_c(\theta_f, \theta_c), \tag{2}$$

$$L_d(\theta_f, \theta_d) = -\sum_{i=1}^{N} d_i * log(s_i) + \pi_d * ||\theta_d||^2, \tag{3}$$

$$L_c(\theta_f, \theta_c) = -\sum_{i=1}^{n} y_i * log(p_i) + \pi_c * (||\theta_f||^2 + ||\theta_c||^2). \tag{4}$$

Here d_i refers to the actual domain class and y_i represents the real polarity. s_i is the predicted domain and p_i is the predicted sentiment. π represents the L2-regularisation term for the class and domain discriminator with underscore c and d, respectively. Last, n equals the source domain sample size and N is the total sample size of the source and target domain data combined. As described, both the source and target aspects are fed into the domain discriminator and only the source instances are passed into the class discriminator. As one can notice, this function is now minimised when the loss of the domain discriminator is maximised. This min-max situation resolves to:

$$\hat{\theta}_d = argmax_{\theta_d} L'_{c,d}(\hat{\theta}_f, \hat{\theta}_c, \theta_d) \tag{5}$$

$$(\hat{\theta}_f, \hat{\theta}_c) = argmin_{\theta_f, \theta_c} L'_{c,d}(\theta_f, \theta_c, \hat{\theta}_d) \tag{6}$$

At this saddle point, the parameters of the domain discriminator, θ_d (Eq. 3), minimise the domain classification error. Secondly, θ_c and θ_f are computed to optimise Eq. 2 by minimising the sentiment prediction loss and maximising the domain classification error. The hyperparameter λ regulates the balance and trade-off between both goals.

The original DANN paper [7] implements Stochastic Gradient Descent (SGD) optimisation. However, the state-of-the-art image classifying model proposed in [19] shows that utilising the faster momentum method [16] instead of SGD also produce accurate results. In each iteration, the parameters of the neural network will be updated according to this method:

$$v_t \longleftarrow \gamma * v_{t-1} + \eta * \nabla_{\theta_k} L(\theta_k) \tag{7}$$

$$\theta_k \longleftarrow \theta_k - v_t. \tag{8}$$

Here, the hyperparameters are the learning rate, η, and momentum factor, γ. In addition, the parameter θ_k represents the weights and biases for the domain discriminator, the feature extractor, and the class discriminator, with $k = d$, $k = f$, and $k = c$, respectively. Last, L represents the corresponding loss function.

4.2   Training Procedure

After constructing the feature representations by the feature extractor, both the source and target domain aspects are passed into the domain discriminator. But, only the source instances are fed into the class discriminator. In our research, the aspects of the target domain also contain a sentiment polarity, but this information is not used in the training and remains unknown to the model up until the moment of testing. The performance of the DAT-LCR-Rot-hop++ is evaluated based on this testing accuracy. The benefit of being able to employ a model, which is trained only on the labels of a source domain, on a target domain gives the DANN approach an advantage over other methods.

The weights and biases are improved using the combined loss function, given by Eq. 2. This equation includes the $-\lambda$ multiplication in order to create sentiment discriminative and domain indiscriminate features. The domain discriminator uses ascending gradient to maximise this loss function, whereas the feature

extractor and the class discriminator use descending gradient to minimise it. The exact training procedure is shown in Algorithm 1. The stopping condition is $max(acc_{t-1}, acc_{t-2}) - acc_{t-3} > \epsilon$, which specifies that if the maximum of the accuracy of the previous epoch and the epoch before minus the loss of three epochs ago is larger than ϵ continue with training. In other words, we continue if there is still a significant improvement.

Algorithm 1. Training procedure of Domain-Adversarial Learning

---

$\epsilon = 0.50\%$
while stopping condition is not met do **do**
    for each epoch **do**
        for each iteration i **do**

– Sample approximately identical percentage batch of source domain, $S(x_i, y_i, d_i)$, and target, $T(x_i, y_i, d_i)$, data. n denotes the source domain batch size and N is the total batch size (source and domain combined).
– Feed input into feature extractor to obtain instance representations (r).
– Pass both $S(x_i, y_i, d_i)$ and $T(x_i, y_i, d_i)$ into domain discriminator and forecast the actual domain d_i. The predicted domain is defined as s_i. Afterwards update the parameters of the discriminator, θ_d, according to the loss function with ascending gradient:

$$\nabla_{\theta_d}[-\lambda(-\frac{1}{N}\sum_{i=1}^{N}d_i * \log(s_i) + \pi_d * ||\theta_d||^2)]$$

– Last, feed $S(x_i, y_i, d_i)$ in class discriminator and predict the real label, y_i. The predicted sentiment is represented by p_i. Finally, adjust the parameters of both the feature extractor, θ_f, and sentiment classifier, θ_c, using the previously estimated domain discriminator parameters, $\hat{\theta}_d$, with descending gradient:

$$\nabla_{\theta_f, \theta_c}[-\lambda(-\frac{1}{N}\sum_{i=1}^{N}d_i * \log(s_i) + \pi_d * ||\theta_d||^2]) +$$

$$-\frac{1}{n}\sum_{i=1}^{n}y_i * \log(p_i) + \pi_f * (||\theta_f||^2 + ||\theta_c||^2)]$$

        end for
    end for
end while

The other hyperparameters besides λ in DAT-LCR-Rot-hop++ are the learning rates, η_k, the momentum terms, γ_k, the L2-regularisation terms, π_k, and the dropout rate. $k = d$ for the domain discriminator and $k = c$ for the feature extractor and class discriminator. First, η determines the rate at which the momentum optimiser converges. In addition, γ determines the influence of past gradient values on the current instance. Furthermore, π reduces overfitting. Fourth, as previously described, λ is a parameter that balances the trade-off

between the discriminative objectives of the class and domain discriminator. Last, the dropout probability regulates the number of layer outputs to be randomly dropped from the network in order to prevent overfitting.

Because the dropout rate does not differ between the methods proposed in [32] and [33], this variable is kept at 0.3 in this research. The remaining hyperparameters (η_k, γ_k, π_k, and λ) are determined by a Tree-structured Parzen Estimator (TPE), which replaces the distribution of the initial observations with a non-parametric distribution by applying a threshold that splits the observations based on different densities [1].

As in the research performed in [32] and [33], the dimension of the word embeddings, $1 \times d$, is equal to 1×768. For convenience, the number of nodes in the Bi-LSTMs, bilinear, and hierarchical attention layer are the same as in [32]. These are 300, 600, and 600, respectively. The number of hidden layers and cells in both the class and domain discriminator is optimised by TPE. The weights of the layers are initialised randomly using a normal distribution with a zero mean. The biases are set to zero at the start.

After the hyperparameters are initialised, DAT-LCR-Rot-hop++ is trained on the training set. The sentiment accuracy of the validation set is used to decide which combination of hyperparameters achieves the best performance. We decided to let the program run 15 times for each source-target domain combination with different settings for the structure and the hyperparameters. Each run includes 50 epochs. The hyperparameter fine-tuning occurs twice. In the first step, λ is excluded, because we want to show the effect of λ on the cross-domain performance of the model. A higher λ should increase the domain-invariance of the features. As a result, λ will first be set to a value of 1.0 [8] in order to find the optimal values for the other parameters. After that the influence of λ is analysed, all hyperparameters, including λ, are fine-tuned to define the best possible configuration. This setting is applied for the final training optimisation with a maximum of 200 epochs.

5   Evaluation

In Sect. 5.1, we first describe the influence of λ on the performance of DAT-LCR-Rot-hop++. Then in Sect. 5.2, the results for the final optimisation are shown.

5.1   Impact of λ

First, the optimal number of hidden layers and neurons together with the values for the hyperparameters are computed. DAT-LCR-Rot-hop++ is run for 7 incrementing values of λ with these settings, starting from 0.5 up until 1.1 with a step of 0.1 for each domain combination (Fig. 2). The impact of the balance hyperparameter λ is visualised by the six graphs that follow in this section. In these graphs, the dark blue line represents the labeling accuracies of the test set of the target domain, while the light orange line shows the base performance

when the majority group of the test sample was selected. The model uses a maximum of 50 epochs.
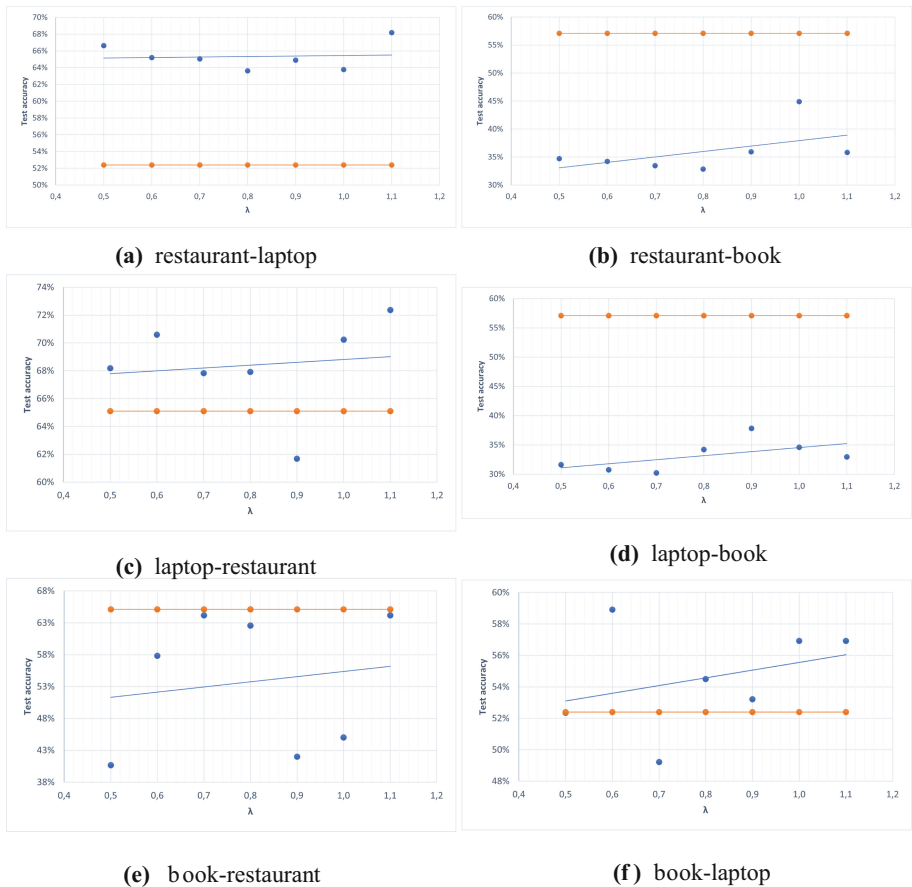

(a) restaurant-laptop

(b) restaurant-book

(c) laptop-restaurant

(d) laptop-book

(e) book-restaurant

(f) book-laptop

Fig. 2. Labeling accuracies for different values of λ, where the dark blue line is used for the target domain classification by our model and the light orange line for the target domain classification by the majority classifier. (Color figure online)

When analysing Fig. 2a and 2c, we observe that the classifying accuracy for the restaurant-laptop and laptop-restaurant domain combination is significantly higher than for the other four. Since the similarities between laptops and restaurants do not seem more prevalent than those between books and laptops, this might come across as a surprising result. However, both the laptop and restaurant domain are taken from the SemEval 2014 dataset [25] while the book domain is retrieved from the ALT 2019 [17]. First of all, these datasets share common context and target text with words such as "service" and "quality". Whereas the ALT 2019 dataset contains these target words 6 and 0 times, respectively,

the words occur 59 and 85 times in the laptop set and 420 and 85 times in the restaurant domain, respectively. In addition, the (digital) language might have changed throughout these 5 years. Last, the fraction of neutral aspects in the book data test set is significantly higher than the neutral percentage in the training sets of both the restaurant and laptop domain with a percentage of 63.1 as compared to 17.7, and 19.8, respectively. This causes emotional phrases, for example "awesome", to appear 5 times in the book domain as compared to 30 and 16 times in the laptop and restaurant domain, respectively. On these grounds, it is expected that the predicting score of the book domain in combination with either laptop or restaurant will result in lower scores compared to laptop-restaurant or restaurant-laptop.

Furthermore, the accuracy of book as a target domain is worse than applying book as a source domain. Because each domain has a disproportionate training set, it causes the neurons of the model to start with predicting the sentiment that occurs most often, especially in the early iterations. So, for both the restaurant and laptop domain this results in predicting a positive sentiment, which leads to a low score for the book target domain. On the other hand, using book as a source domain does lead to acceptable performance. Not surprisingly, most neutral aspects are correctly classified in the book-restaurant and book-laptop combinations, with an average accuracy of 65% and 81%, respectively. Next, DAT-LCR-Rot-hop++ focuses on the second largest polarity percentage, the positive sentiments, which is the majority in both the restaurant and laptop domain. The drawback of this is the bad score for the negative polarities with an average accuracy of 3.2% and 0%, respectively.

When looking at Fig. 2a, we observe a scattered graph with an almost flat regression line. Accordingly, the coefficient of the OLS slope is 0.61, which means that increasing λ with 1.0 increases the testing accuracy by 61%. The same holds for Fig. 2c, which has a slope of 2.04. One reason for this could be the previously mentioned overlap between the restaurant and laptop domain, thereby decreasing the difficulty of the cross-domain task and hence, making λ less important. Besides this result, the restaurant-laptop combination does beat the base performance line at 52% with an average of 65%. The same applies to the laptop-restaurant for which the observation at $\lambda = 0.9$ appears to be an outlier. In contrast to the flat regression lines of the restaurant-laptop and laptop-restaurant, restaurant-book depicts a clear positive linear trend with a slope of 9.73 in Fig. 2b. The same applies to the graph in Fig. 2d, which has a slope of 6.91, showing the effect of λ.

As previously stated, employing book as the target domain produces a poor labeling accuracy. Especially, the positive outlier in Fig. 2b is an observation that proves the effect of the disproportionate data sets. Only during this run, DAT-LCR-Rot-hop++ was capable of predicting neutral sentiments correctly, which results in a neutral accuracy of 26%. As compared to a maximum of 5% for the other runs with book as a target domain. The competence to classify the neutral aspects precisely immediately leads to a significantly better performance with an accuracy of 45%, as compared to approximately 35%. The disproportionate

sets cause DAT-LCR-Rot-hop++ to overfit on the training set and focus only on the major two polarities, resulting in low scores for the restaurant-book and laptop-book combination. Whereas there is a clear ascending performance for the training set, reaching percentages up until 92%, the maximum accuracy of the target domain is reached after approximately 100 epochs. The statistic then moves around this number with some large outliers into both directions, but never really improving. After some time, the accuracy starts to drop. The model becomes too much specified towards the information of the training set.

Both book-restaurant and book-laptop in Fig. 2e and 2f, provide an ascending line with a slope coefficient of 7.04 and 4.88, respectively. However, the data points in Fig. 2e are scattered, causing a standard deviation of 10%. Therefore, this positive relationship might be questioned in this case.

5.2   Final Optimisation

The final values of the hyperparameters for the final optimisation with 200 iterations are defined in Table 2. Each domain combination is tested for the final prediction using these parameter settings. The results are shown in Table 3. As expected, the accuracies improve for each source and target domain model as compared to the previous run with 50 epochs. The training label accuracy increases from 84% up until 92% for the book-laptop domain. In addition, the maximum testing accuracy of 80% for the restaurant-laptop is a 9 percentage points improvement from the previous 71%. The ratios of correctly predicted polarities follow the previously seen distribution.

Table 2. Values for hyperparameter optimisation.

Hyperparam	re-la	re-bo	la-re	la-bo	bo-re	bo-la
lr_d	0.01	0.03	0.03	0.01	0.01	0.03
$lr_{c,f}$	0.005	0.005	0.03	0.01	0.005	0.03
mom_d	0.90	0.85	0.80	0.90	0.85	0.90
$mom_{c,f}$	0.90	0.90	0.85	0.85	0.85	0.85
$l2 - term$	0.001	0.001	0.001	0.001	0.001	0.0001
structure	2400-600	2400-0	2400-600	2400-600	2400-0	2400-600
λ	1.1	0.8	1.1	0.6	1.1	0.6

"re", "la", and "bo" represent restaurant, laptop, and book domain, respectively

The performance of the restaurant-laptop domain increased significantly, which results in a total test accuracy of 77%. The relevance of not requiring any labeled target data should not be underestimated when comparing it with other research, because this ability reduces labeling costs significantly. Specifically, the outcomes for the book-restaurant are promising. Both domains are not closely related in terms of sentiment distribution, but the model achieves an encouraging test accuracy of 72%, which is an improvement of 8% points

Table 3. Test accuracies for DAT-LCR-Rot-hop++ model.

	test	train	max	base	pos	neu	neg
rest-lapt	77%	87%	80%	52%	94%	38%	83%
rest-book	37%	88%	48%	57%	71%	9%	88%
lapt-rest	74%	83%	78%	65%	90%	9%	85%
lapt-book	42%	90%	52%	57%	86%	11%	72%
book-rest	72%	88%	76%	65%	78%	81%	40%
book-lapt	60%	92%	66%	52%	75%	79%	0%

"base" is the majority classifier in the target domain
test set, "pos", "neu", and "neg" are sentiment labels

as compared to the value after the previous runs. Interestingly, the fraction of correctly labeled sentiments is more balanced, instead of one polarity that is driving the results.

6   Conclusion

The prominent role of digital online media increases the relevance of ABSA, and in particular ABSC. Since obtaining labeled target data is extremely costly, new models should be developed that can be employed on a variety of domains, a concept known as transfer learning. [8] introduces a method known as DANN, which is a specification of GAN as defined by [9] for transfer learning. The major benefit of this approach is the fact that it does not need any labeled target data at all.

The state-of-the-art LCR-Rot-hop++ structure from [32] forms the basis of our proposed DAT-LCR-Rot-hop++, which adds an adversarial component based on DANN. It consists of a domain discriminator, a class discriminator, and a GRL module. The GRL module reverses the loss of the domain discriminator before back-propagation, which enforces the earlier layers to generate domain-invariant features. At the same time, the class discriminator is trained on the labels of the source domain. This results in label-discriminative features.

Increasing λ (domain invariance) improves, in general, the performance of the target, especially for domains that are distant. Furthermore, the model is not able to correctly predict the sentiments of the restaurant-book and laptop-book models. The benchmark of predicting the majority class of the test set is not reached for both domain combinations. This is the outcome of multiple factors of which the most essential one is the disproportionate data set in terms of polarity distribution of the three domains. The high percentage of neutral aspects in the book domain causes the model to perform poorly. Nevertheless, the produced models are better than a random classifier, which shows that even in these extreme conditions our approach is able to detect useful signals.

On the other hand, the accuracy score for the restaurant-laptop, laptop-restaurant, and book-restaurant all exceed 72%. So in half of the cases, DAT-LCR-Rot-hop++ is able to properly classify polarities, but it depends on which

combination of domains is used. Domains with similar polarity distributions seem to benefit the most from the proposed approach.

In order to further improve our method, we propose two adaptions. The first one covers the difficulty of predicting three kinds of classes. We would like to investigate the performance of the model for only the binary case. Classifying neutral aspects appears to be a harsh task for the neural network, looking at the results, as this class is poorly represented in two out of our three domains. Our second extension includes transforming the LCR-Rot-hop++ component. In some cases, the model becomes overfitted on the training data, so a less complicated feature extractor might help prevent this. One way to reduce the complexity is to reduce the dimensionality of the internal network layers.

References

1. Bergstra, J., Bardenet, R., Bengio, Y., Kégl, B.: Algorithms for hyper-parameter optimization. In: 25th Annual Conference on Neural Information Processing Systems (NIPS 2011), pp. 2546–2554. Curran Associates (2011)
2. Brauwers, G., Frasincar, F.: A survey on aspect-based sentiment classification. ACM Comput. Surv. **1**, 35 (2022)
3. Chapman, B.: GameStop: reddit users claim victory as \$13bn hedge fund closes position, accepting huge losses (2021). https://www.independent.co.uk/news/business/gamestop-share-price-reddit-hedge-fund-melvin-capital-b1793543.html
4. Chen, Y., Tong, Z., Zheng, Y., Samuelson, H., Norford, L.: Transfer learning with deep neural networks for model predictive control of HVAC and natural ventilation in smart buildings. J. Clean. Prod. **254**, 119866 (2020)
5. Ciresan, D., Meier, U., Schmidhuber, J.: Multi-column deep neural networks for image classification. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2012), vol. 1, pp. 3642–3649. IEEE Computer Society (2012)
6. Devlin, J., Chang, K., Lee, K., Huang, D., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT 2019), pp. 4171–4186. ACL (2019)
7. Ganin, Y., Lempitsky, V.: Unsupervised domain adaption by backpropagation. In: 32nd International Conference on Machine Learning (ICML 2015), vol. 37, pp. 1180–1189. PMLR (2015)
8. Ganin, Y., et al.: Domain-adversarial training of neural networks. J. Mach. Learn. Res. **17**, 59:10–59:35 (2016)
9. Goodfellow, I., et al.: Generative adversarial networks. In: 28th Annual Conference on Neural Information Processing Systems (NIPS 2014), pp. 2672–2680 (2014)
10. Hendricks, D.: Complete history of social media: then and now (2013). https://smallbiztrends.com/2013/05/the-complete-history-of-social-media-infographic.html
11. Hong, W., Wang, Z., Yang, M., Yuan, J.: Conditional generative adversarial network for structured domain adaption. In: 8th IEEE International Conference on Computer Vision and Pattern Recognition (CVPR 2018), vol. 12, pp. 1335–1344. IEEE (2018)
12. Hong, Y., Zhou, W., Zhang, J., Zhu, Q., Zhou, G.: Self-regulation: employing a generative adversarial network to improve event detection. In: 56th Annual Meeting

of the Association for Computational Linguistics (ACL 2018), pp. 515–526. ACL (2018)

13. Johansson, F., Shalit, U., Sontag, D.: Learning representations for counterfactual inference. In: 33rd International Conference on Machine Learning (ICML 2016). JMLR Workshop and Conference Proceedings, vol. 48, pp. 3020–3029. JMLR (2016)

14. Kamath, S., Gupta, S., Carvalho, V.: Reversing gradients in adversarial domain adaption for question deduplication and textual entailment tasks. In: 57th Annual Meeting of the Association for Computational Linguistics (ACL 2019), pp. 5545–5550. ACL (2019)

15. Lai, S., Xu, L., Liu, K., Zhao, J.: Recurrent convolutional neural networks for text classification. In: 29th Conference on Artificial Intelligence (AAAI 2015), vol. 29, pp. 2267–2273. AAAI Press (2015)

16. Liu, C., Belkin, M.: Accelerating SGD with momentum for over-parameterized learning. In: 8th International Conference on Learning Representations (ICLR 2020). OpenReview.net (2020)

17. Álvarez-López, T., Fernández-Gavilanes, M., Costa-Montenegro, E., Bellot, P.: A proposal for book oriented aspect based sentiment analysis: comparison over domains. In: Silberztein, M., Atigui, F., Kornyshova, E., Métais, E., Meziane, F. (eds.) NLDB 2018. LNCS, vol. 10859, pp. 3–14. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-91947-8_1

18. Maat, E.D., Krabben, K., Winkels, R.: Machine learning versus knowledge based classification of legal texts. In: 23rd Annual Conference on Legal Knowledge and Information Systems (JURIX 2010), vol. 223, pp. 87–96. IOS Press (2010)

19. Mauro, M., Mazzia, V., Khalil, A., Chiaberge, M.: Domain-adversarial training of self-attention based networks for land cover classification using multi-temporal sentinel-2 satellite imagery. arXiv preprint arXiv:2104.00564 (2021)

20. Pan, S., Yang, Q.: A survey on transfer learning. IEEE Trans. Knowl. Data Eng. **22**, 1345–1359 (2009)

21. Pang, B., Lee, L.: A sentimental education: sentiment analysis using subjectivity summarization based on minimum cuts. In: 42nd Annual Meeting of the Association for Computational Linguistics (ACL 2004), pp. 271–278. ACL (2004)

22. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up? Sentiment classification using machine learning techniques. In: 2002 Conference on Empirical Methods in Natural Language Processing 2002 (EMNLP 2002), pp. 79–86. ACL (2002)

23. Pontiki, M., Galanis, D., Papageorgiou, H., Androutsopoulos, I., Manandhar, S.: SemEval-2015 task 12: aspect based sentiment analysis. In: 9th International Workshop on Semantic Evaluation (SemEval 2015), pp. 486–495. ACL (2015)

24. Pontiki, M., et al.: SemEval-2016 task 5: aspect based sentiment analysis. In: 10th International Workshop on Semantic Evaluation (SemEval 2016), pp. 19–30. ACL (2016)

25. Pontiki, M., Galanis, D., Pavlopoulos, J., Papageorgiou, H., Androutsopoulos, I., Manandhar, S.: SemEval-2014 task 4: aspect based sentiment analysis. In: 8th International Workshop on Semantic Evaluation (SemEval 2014), pp. 27–35. ACL (2014)

26. Schouten, K., Frasincar, F.: Survey on aspect-level sentiment analysis. IEEE Trans. Knowl. Data Eng. **28**, 813–880 (2016)

27. Schouten, K., Frasincar, F.: Ontology-driven sentiment analysis of product and service aspects. In: Gangemi, A., et al. (eds.) ESWC 2018. LNCS, vol. 10843, pp. 608–623. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-93417-4_39

28. Taboada, M., Brooke, J., Tofiloski, M., Voll, K., Stede, M.: Lexicon-based methods for sentiment analysis. Comput. Linguist. **37**, 267–307 (2011)
29. Tankovska, H.: Social media - statistics and facts (2021). https://www.statista.com/topics/1164/social-networks/
30. Thet, T., Na, J., Khoo, C.: Aspect-based sentiment analysis of movie reviews on discussion boards. J. Inf. Sci. **36**, 823–848 (2010)
31. Towell, G., Shavlik, J.: Knowledge-based artificial neural networks. Artif. Intell. **70**, 119–165 (1994)
32. Truşcă, M.M., Wassenberg, D., Frasincar, F., Dekker, R.: A hybrid approach for aspect-based sentiment analysis using deep contextual word embeddings and hierarchical attention. In: Bielikova, M., Mikkonen, T., Pautasso, C. (eds.) ICWE 2020. LNCS, vol. 12128, pp. 365–380. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-50578-3_25
33. Wallaart, O., Frasincar, F.: A hybrid approach for aspect-based sentiment analysis using a lexicalized domain ontology and attentional neural models. In: Hitzler, P., et al. (eds.) ESWC 2019. LNCS, vol. 11503, pp. 363–378. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-21348-0_24
34. Wang, F., Zhang, Q.: Knowledge-based neural models for microwave design. IEEE Trans. Microw. Theory Tech. **45**, 2333–2343 (1997)
35. Wang, Z., Huang, M., Zhao, L., Zhu, X.: Attention-based LSTM for aspect-level sentiment classification. In: 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP 2016), pp. 606–615. ACL (2016)
36. Yanase, T., Yanai, K., Sato, M., Miyoshi, T., Niwa, Y.: bunji at SemEval-2016 task 5: neural and synctactic models of entity-attribute relationship for aspect-based sentiment analysis. In: 10th International Workshop on Semantic Evaluation (SemEval 2016), pp. 289–295. ACL (2016)
37. Yosinski, J., Clune, J., Bengio, Y., Lipson, H.: How transferable are features in deep neural networks? In: 27th Annual Conference on Neural Information Processing Systems (NIPS 2014), vol. 27, pp. 3320–3328. Curran Associates (2014)
38. Yuan, J., Zhao, Y., Qin, B., Liu, T.: Learning to share by masking the non-shared for multi-domain sentiment classification. arXiv preprint arXiv:2104.08480 (2021)
39. Zhang, W., Ouyang, W., Li, W., Xu, D.: Collaborative and adversarial network for unsupervised domain adaption. In: 2018 Conference on Computer Vision and Pattern Recognition (CVPR 2018), pp. 3801–3809. IEEE (2018)
40. Zhang, Y., Qiu, Z., Yao, T., Liu, D., Mei, T.: Fully convolutional adaption networks for semantic segmentation. In: 2018 International Conference on Computer Vision and Pattern Recognition (CVPR 2018), pp. 6810–6818. IEEE (2018)
41. Zheng, L., Zhang, Y., Wu, Y., Wei, Y., Yang, Q.: End-to-end adversarial memory network for cross-domain sentiment classification. In: 26th International Joint Conference on Artificial Intelligence (IJCAI 2017), pp. 2237–2243. IJCAI (2017)
42. Zheng, S., Xia, R.: Left-center-right separated neural network for aspect-based sentiment analysis with rotatory attention. arXiv preprint arXiv:1802.00892 (2018)