Labels are not necessary: Assessing peer-review helpfulness using domain adaptation based on self-training

Chengyuan Liu
North Carolina State University
cliu32@ncsu.edu

Divyang Doshi
North Carolina State University
ddoshi2@ncsu.edu

Muskaan Bhargava
North Carolina State University
mbharga5@ncsu.edu

Ruixuan Shang
University of North Carolina at Chapel Hill
rshang@unc.edu

Jialin Cui
North Carolina State University
jcui9@ncsu.edu

Dongkuan Xu
North Carolina State University
dxu27@ncsu.edu

Edward Gehringer
North Carolina State University
efg@ncsu.edu

Abstract

A peer-assessment system allows students to provide feedback on each other's work. An effective peer assessment system urgently requires helpful reviews to facilitate students to make improvements and progress. Automated evaluation of review helpfulness, with the help of deep learning models and natural language processing techniques, gains much interest in the field of peer assessment. However, collecting labeled data with the "helpfulness" tag to build these prediction models remains challenging. A straightforward solution would be using a supervised learning algorithm to train a prediction model on a similar domain and apply it to our peer review domain for inference. But naïvely doing so can degrade the model performance in the presence of the distributional gap between domains. Such a distributional gap can be effectively addressed by Domain Adaptation (DA). Self-training has recently been shown as a powerful branch of DA to address the distributional gap. The first goal of this study is to evaluate the performance of self-training-based DA in predicting the helpfulness of peer reviews as well as the ability to overcome the distributional gap. Our second goal is to propose an advanced self-training framework to overcome the weakness of the existing self-training by tailoring knowledge distillation and noise injection, to further improve the model performance and better address the distributional gap.

1 Introduction

Peer review is a learning tool that enables students to evaluate their peers' assignments or projects (Gamage et al., 2021; Topping, 2009; Li et al., 2019). It can help instructors enhance their teaching (Çevik et al., 2015; Gamage et al., 2021), and allow students to develop skills in assessing and providing feedback to others. Figure 1 illustrates the steps of the peer review process. It starts with the authors submitting their work. The peers then evaluate the work and provide both textual feedback and numerical scores. The author assesses the feedback and tends to accept only the helpful reviews to make further revisions (Lundstrom and Baker, 2009). The instructors can refer the numerical scores provided by the reviewers to give the final grades. Therefore, identifying helpful peer reviews can enhance the benefits to students from the peer-review process (Nelson and Schunn, 2009; Ramachandran et al., 2017). Automatic recognition of peer-review helpfulness has been studied limitedly with the help of deep learning models and natural language processing (Xiong and Litman, 2011b;Xiao et al., 2022). However, in order to create a reliable model that can accurately predict helpfulness, a considerable amount of peer-review data labeled with helpfulness is required (Chapelle et al., 2009). The students receiving the reviews are the most suitable individuals to label the data, but the difficulty in collecting labeled reviews from stu-

Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023), pages 173–183
July 13, 2023

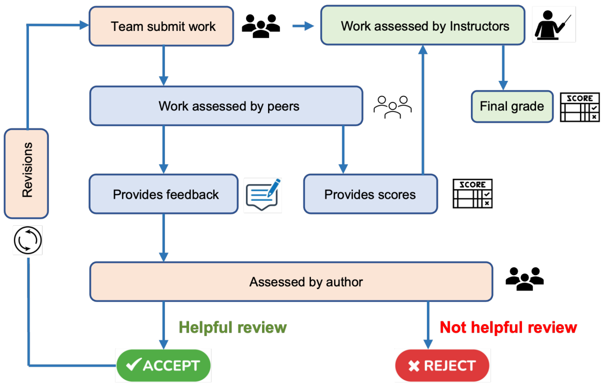

Figure 1: Peer review process flowchart. The pipeline involves the evaluation of the feedback from peers by the author. Only the helpful review is accepted and taken into account for further revisions.

dents poses a challenge. Moreover, the subjective nature of "helpfulness" creates ambiguity, making it challenging to achieve a consensus in a team on whether a review is helpful. As a result, obtaining sufficient labeled data to develop a robust model for predicting helpfulness remains a significant obstacle.

A straightforward solution to overcome the challenges of collecting labeled data is to adapt a model trained on a pre-existing labeled dataset from a similar domain that includes "helpfulness" tags to our peer review domain. Specifically, we can train the prediction model on a **"source domain"** labeled data following the supervised manner, and generate "helpfulness" prediction on the **"target domain"** unlabeled data from our peer reviews. However, the discrepancies in data distribution between the source and target domains, i.e., **domain shift** (Li et al., 2020;Wang and Breckon, 2020), can cause the model's performance to degrade on unseen target domain data.

In this paper, in order to address the domain shift issue, we propose to apply **Self-training** (a.k.a., Pseudo-labeling) (Zou et al., 2019; Lee, 2013;Feng et al., 2021;Mei et al., 2020;Yu et al., 2021), as a promising technique in Domain Adaptation (DA) ((Ben-David and Urner, 2012; Liu et al., 2021; Zou et al., 2019)). Self-training-based Domain Adaptation aims to transfer knowledge learned from the source domain to the target domain, by involving the unlabeled data from the target domain in the model training. We hypothesize that learning from the unlabeled data can enhance the generalization ability, and facilitate the effective knowledge transfer across domains. This hypothesis will be validated through the experiment results.

Our proposed approach for domain adaptation using self-training follows the "student-teacher model" framework (Pu and Li, 2023). As shown in Figure 3, the student and teacher models will constantly exchange their roles during the iterative process, and the student model will continuously learn from the pseudo labels predicted by the teacher model. Self-training helps to overcome the domain shift between the source and target domains (Liu et al., 2021). As a novelty, our study also proposes an advanced self-training framework that utilizes knowledge distillation (Hinton et al., 2015) and noise injection (Xie et al., 2020) techniques to overcome some weaknesses of the traditional self-training, and further improve the adaptation performance. By incorporating knowledge distillation, the student model can better mimic the teacher model and break through the limitation of only being able to learn from the "hard labels" provided by the teacher model. Additionally, the incorporation of noise injection enables the student model to outperform the teacher model by learning from the augmented data, which is beyond what the teacher model predicts.

The contributions are summarized as follows:

– We propose the use of self-training-based domain adaptation to predict peer review helpfulness, which overcomes the challenge of collecting labeled data and mitigates the domain shift issue.

– We improve self-training by tailoring knowledge distillation techniques and utilizing soft labels to provide more comprehensive knowledge for the student model to learn from the teacher model.

– We improve self-training by introducing noise during the student model training phase, enabling the student model to learn beyond the predictions generated by the teacher model.

2 Related Work

2.1 Peer Review Helpfulness Prediction

Previous peer-review research has not paid much attention to helpfulness prediction, with only a few studies utilizing NLP techniques to identify key features in review comments to evaluate the quality. Xiong and Litman (2011a) conducts a pioneering study on predicting peer-review helpfulness and suggests that techniques used in other domains can be applied to the peer-review domain. Zingle et al. (2019) describes a method for automatically detecting *suggestions* in review text. Xiao et al. focus on detecting *problem statements* which point out the problems that need to be addressed in review comments.

However, there is no study that directly investigates predicting helpfulness based on the semantics of the review content. The lack of labeled training data also poses a challenge to building such a prediction model, due to the subjective nature of helpfulness and controversies surrounding its definition. Xiong and Litman (2011b) reports that there is a great deal of variation among students and even domain experts in terms of "what constitutes a helpful comment."

Fortunately, several researchers (Tsur and Rappoport, 2009; Qu et al., 2020; Yang et al., 2015) have explored predicting the helpfulness of online product reviews, which can be conveniently labeled with "helpfulness" through user voting from online shopping platforms. In this study, we adapt the task of predicting the helpfulness of online product reviews to our academic peer reviews, drastically reducing the need for collecting peer-review labeled data.

2.2 Domain Adpatation

Training models on the "source domain"(with labeled data) and testing them on the "target domain" (without labeled data) using supervised learning algorithms often fail due to the distributional gap between the two domains, commonly known as domain shift (Long et al., 2015).

Domain adaptation (DA) aims to alleviate the effect of domain shift. Various methods have been proposed to mitigate that by aligning the source and target domain in the feature space. These approaches explicitly align their statistics or use adversarial learning. For instance, Glorot et al. (2011) proposed an autoencoder-based domain adaptation network, which extracts high-level representations from both source and target domain data. They then trained a linear classifier to learn from the source data's extracted features and applied it to the target data. Long et al. (2015) used a deep neural network to learn transferable features across domains by adding multiple adaptation layers to the task-specific representations. They match the marginal distributions of both domains. Furthermore, Ganin and Lempitsky (2015) proposed an adversarial-based domain adaptation approach that adds an effective Gradient Reversal Layer (GRL) to the model, inspired by *Generative Adversarial Networks* (Goodfellow et al., 2014), to match the domain gap.

Despite the success of the existing approaches, Ben-David and Urner (2012) highlighted the difficulty of applying the above feature-adaptation-based approaches in DA and suggested that none of those methods have the capacity to generalize well to the unlabeled target domain data. In this study, we propose to use self-training (a.k.a, *pseudo-labeling*) as a promising alternative to the feature-adaptation approaches to better handle the domain shift.

2.3 Self-training

Self-training is a popular technique in semi-supervised learning, where a supervised method is applied for classification or regression tasks in a semi-supervised manner. In self-training, the model is trained on a small amount of labeled data, then it generates predictions on the unlabeled data, which are adopted as pseudo-labels. The model is retrained on the combination of both labeled data and pseudo-labeled data, and the process iterates until convergence.

In pioneering work, Lee (2013) first introduces the classical pseudo-labeling method, which differs from the self-training framework in that the model is not retrained after each pseudo-labeling. He et al. (2020) successfully applies the self-training framework in NLP tasks such as machine translation and text summarization, also provides a comprehensive evaluation of its effectiveness. Another approach proposed by Pu and Li (2023) is the self-training framework with a "student-teacher model", in which a teacher model assigns pseudo-labels to unlabeled data, and a student model is trained on the combined dataset iteratively. However, the vanilla self-training suffers from certain limitations of the student model's learning abilities, which we defined as "inability to learn sufficiently from the teacher model" and "inability to learn beyond the teacher model".

To address these limitations, we propose applying knowledge distillation and noise injection to the self-training framework, which ensures a well-performing student model. Our approach improves the student model's learning ability, achieving decent results over the traditional self-training approach.

3 Methodology

3.1 Self-training for Domain Adaptation

Self-training for domain adaptation is a bit different from the traditional single-domain self-training approach, the workflow is illustrated in Figure 3 and formulated using the following steps:
Requirements: Source-domain labeled dataset $D_{SL} = \left\{\left(x_i^L, y_i\right)\right\}_{i=1}^{N_{sl}}$ and target-domain unlabeled dataset $D_{TU} = \left\{\left(x_j^U\right)\right\}_{j=1}^{N_{tu}}$ where N_{sl} and N_{tu} stands for the number of samples in source and target dataset respectively; x_i^L and x_j^U are the vector representations of each review text; and y_i stands for the one-hot encoding label for source domain labeled data.
Steps:

1. To initiate the self-training process, a teacher model $f_\mathcal{T}\left(\theta_*\right)$ (e.g., a BERT-based language classification model (Devlin et al., 2019)) is trained on the labeled dataset from the source domain, to minimize the cross-entropy loss using Equation 1.

$$\frac{1}{N_{sl}}\sum_{i=1}^{N_{sl}} CE(y_i, f_\mathcal{T}\left(x_i^L, \theta\right)) \tag{1}$$

2. The teacher model is then used to generate pseudo-labels on the unlabeled dataset from the target domain, as shown in Equation 2.

$$\hat{y_j} = f_\mathcal{T}\left(x_j^U, \theta_*\right), \forall j \in [1, N_{tu}] \tag{2}$$

3. A student model $f_s\left(\theta'_*\right)$ (e.g., BERT-based language classification model) is then learned to minimize the cross entropy loss on a combined dataset $D_C = \{(x_c)\}_{c=1}^{N_c}$, which includes the source domain labeled data D_{SL} and target domain pseudo-labeled data $\hat{D_{TU}}$. The loss is calculated using Equation 3.

$$\frac{1}{N_c}\sum_{c=1}^{N_c} CE(y_c, f_s\left(x_c, \theta'\right)) \tag{3}$$

where $N_c = N_{sl} + N_{tu}$, (x_c, y_c) represents (x_i, y_i) and $(x_j, \hat{y_j})$ for the source labeled set and the target pseudo-labeled set, respectively.

3.2 Knowledge Distillation — "Student Learns More From Teacher"

Knowledge Distillation (KD) is a technique for compressing a model by using a more complex teacher model that has already been trained to guide a smaller, less-complex student model. This is done to maintain the accuracy of the original teacher model while reducing the model size and computational resources required (Hinton et al., 2015).

In traditional classification, the model aims to map input features to the one-hot labels, which only provide class information. However, with KD, the teacher model can generate a continuous distribution of class labels (i.e., soft labels) for each sample, allowing for more information to be used. The student model is then trained to closely match the output distribution of the teacher model.

Specifically, KD employs softmax probability to generate soft labels. In contrast, traditional classification tasks use cross-entropy as the loss function, with hard one-hot labels as targets. However, as highlighted by Hinton et al. (2015), this approach can result in the loss of valuable information on the similarity between and within classes. By using the probability output from the softmax layer instead, KD is able to retain more information.

Incorporating the KD technique into our self-training framework aims to improve the performance of the student model by acquiring additional knowledge from the pseudo-labels generated by the teacher model. Figure 3 illustrates the process of knowledge distillation in self-training. In this process, we retained both the hard and soft pseudo-labels generated by the teacher model to preserve an adequate amount of information. Consequently, we substituted the conventional cross-entropy loss function with the KD loss function (Hinton et al., 2015) as represented in Equation 4.

$$L = -\sum_{i=1}^{K} p_i^{hard}\log q_i + \sum_{i=1}^{K} p_i^{soft}\log(\frac{p_i^{soft}}{q_i}) \tag{4}$$

The first segment of the equation calculates the cross-entropy loss between the hard pseudo-labels p_i^{hard} (one-hot encoding), which are generated by the teacher model and represented through one-hot encoding, and the soft output q_i produced by the student model. The latter part computes the Kullback–Leibler divergence (Wikipedia contributors, 2023) between the soft pseudo-labels p_i^{soft} from the teacher model and the output q_i of the student model. Our objective is to account for both the

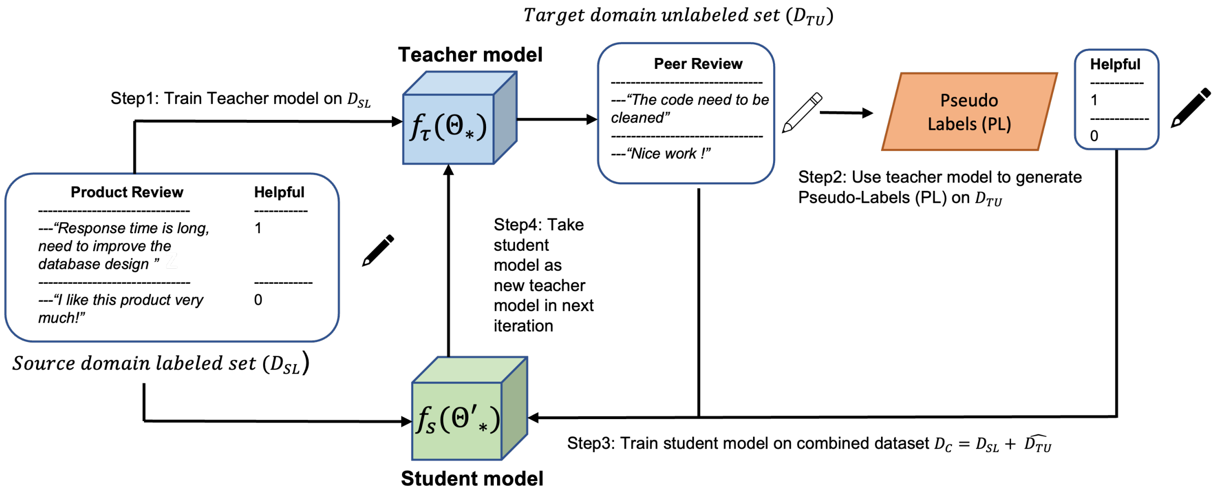

Figure 2: Self-training pipeline for peer review helpfulness detection across domains. A "Teacher model" will be trained on the labeled data from the source domain. Then a "Student model" will be trained using both the labeled data from the source domain and the pseudo-labeled data from the target domain labeled by the teacher model. The trained "Student model" will be used as the new "Teacher model" in the next iteration.

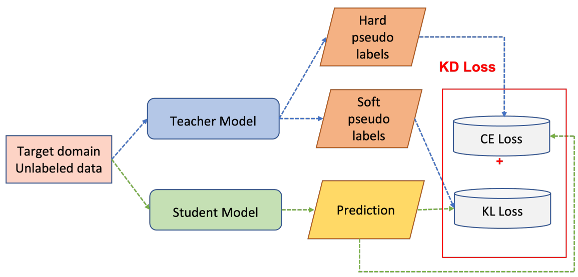

Figure 3: Schematic diagram of the KD loss computation in single self-training iteration

hard and soft pseudo-labels' information while calculating the loss.

3.3 Noise Injection – "Student Learns Beyond Teacher"

The use of Knowledge Distillation enables the student model to learn more information from the soft labels. However, it is crucial to acknowledge that the primary objective of employing KD is to train a smaller and more efficient student model that has the same capabilities as the teacher model. Conversely, in self-training, our goal is to train a superior performing student model. To achieve this, we must ensure that the student model is not less complex than the teacher model and has the ability to capture more variance of the data. Unfortunately, incorporating KD is insufficient to accomplish this.

Noise injection creates a more challenging environment for the student model to learn beyond the predictions. In this study, we utilize data augmentation as the noise injection method in the student model training phase. We implement back-translation (Ng et al., 2019) as a prominent text-augmentation approach on the target domain's pseudo-labeled data. For the augmented data, we keep the same pseudo-labels (both hard and soft). Consequently, this requires the student model to ensure that a translated version of the text yields the same output as the original text, which is also known as *consistency regularization* (Ho et al., 2022). By doing this, we improve the student model by providing augmented data to learn beyond what the teacher model predicts.

4 Experiments and Results

4.1 Datasets

Source Domain Labeled Data. Our source domain labeled data is obtained from the *Amazon Product Review* (Ni et al., 2019), which contains 29 categories of online products. Since the categories' relevance to our peer-review data varies, we conduct experiments on two product categories. The "software" category is chosen, since it is closely related to our peer-review data, as both involve user-experience feedback on developed applications. The "automotive" category is also selected to evaluate whether data from a less-relevant domain would impact the performance of domain adaptation. Additionally, we create two datasets of varying size within each category and investigate how significantly the size affects the performance.

Our objective is to predict binary class labels of

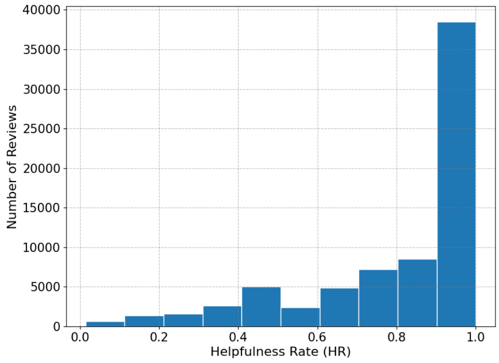

Figure 4: Helpfulness rate distribution of "software" product review. Note in these plots that the majority of the reviews have the "helpfulness ratio" larger than 0.8.

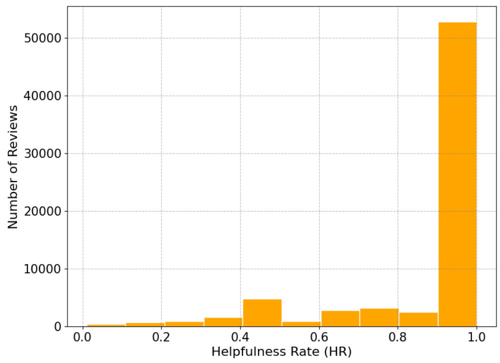

Figure 5: Helpfulness rate distribution of "automotive" product review.

reviews, where "0" represents "not helpful" and "1" represents "helpful". However, the original data contain the "helpfulness" tags, which have been collected through user votes formatted as: "the number of users who find the review helpful out of the total number of users who vote for the review" (e.g., [2,3] implies that out of 3 users who voted on the review, 2 of them rated it as helpful, thus the "helpfulness ratio (hr)" is 2/3). To convert this into binary class labels, we decide to set a threshold for the "helpfulness ratio" and split the data into the two classes of "helpful" and "not helpful". Figures 4 and 5 illustrate the distribution of the "helpfulness ratio" for "software" and "automotive" datasets. To create a clear distinction between the two classes, we choose the reviews with a "helpfulness ratio" above 0.85 as helpful and below 0.35 as unhelpful reviews.

After text cleaning and processing, we collect 500 and 2000 labeled product reviews for each of the two categories. We also ensured that the class labels are evenly distributed.

Target Domain Unlabeled Data The peer review data of the target domain is collected from the Expertiza system (Gehringer et al., 2006), which is a web-based peer review system used in a masters-level computer science class. The system requires students to review assignments from their peers and provide numerical scores and textual feedback. We extract the textual feedback data from the fall semesters of 2017 to 2020, resulting in 24,619 review samples after cleaning and processing.

Target Domain Validation Data. We should also need a validation set from the target domain to assess whether it generalized well by using our proposed self-training approach. However, collecting "helpfulness" tags in our peer review system is challenging. Fortunately, the Expertiza system (Gehringer et al., 2006) provides a way for students to tag the reviews they received as having or not having particular characteristics. These tags identified features such as *contains problem statement* and *contains suggestion*. A study conducted by Xiao et al. (2022) states that these two features are highly correlated with review helpfulness. Therefore, we decide to utilize these tags as a proxy for "helpfulness" tag to create our target domain validation data.

We generated the "helpfulness" label for our validation sets by considering review comments tagged as containing *both* "problem statement" and "suggestion" as "helpful" and those without either of these two as "not helpful". (Comments containing either "problem statement" or "suggestion" tags, but not both, were excluded from the dataset.) The result was a balanced validation set of 7000 reviews, consisting of an equal number of "helpful" and "not helpful" samples.

4.2 Experiment Settings

Supervised Learning Baseline The first baseline method uses a supervised learning approach. We aim to investigate the existence of a domain shift in our task. We applied the pre-trained "bert-base-uncased" model from the Hugging Face library (Wolf et al., 2019) and fine-tuned it on the labeled dataset from the source domain. Then, we validated its performance on the target domain validation set. The domain shift is evaluated by calculating the accuracy score of the model on the validation set. The detailed settings of this baseline

Parameters	Value
Tokenizer	*'bert-based-uncased'*
Classification model	*'bert-based-uncased'*
Number of classes	*2*
Loss function	*Cross-entropy loss*
Optimizer	*Adam*
Dropout	*0.3*
Learning rate	*2e-5*
Epoch	*5*
Batch size	*16*

Table 1: Supervised learning baseline experiment setting

are presented in Table 1.

Self-training Baseline We establish our second baseline as applying the vanilla self-training approach to examine whether learning from the target domain unlabeled data could enhance the performance and address the domain shift. As shown in Figure 3, the training of the teacher model uses the exact same settings as the supervised learning baseline presented in Table 1. Afterward, the self-training loop is initiated, where each loop starts by generating pseudo-labels using the trained teacher model and ends by taking the trained student model as the new teacher model. In the self-training phase, we have set the value of *outer_epoch* to 10, which indicates how many times we will repeat the loop described above. Additionally, we also set the value of *inner_epoch* to 3, which represents the number of training iterations of the student model in each self-training loop.

Our Proposed Approach To overcome the limitations of the self-training, we propose an approach that integrates *knowledge distillation* and *noise injection* in the self-training loop. The core idea behind knowledge distillation is to generate soft pseudo-labels in the form of prediction probabilities to enable student models to learn from additional knowledge. Therefore, in addition to retaining the prediction probabilities from the teacher model, we also replace the cross-entropy loss with the “kd_loss” (defined in Equation 4) for training the student model. However, we continue to use the cross-entropy loss for training the teacher model with hard labels. Consequently, the general loss function of both the student and teacher models can be formulated as follows:

$$loss = \alpha \times KL\_loss + (1 - \alpha) \times CE\_loss \quad (5)$$

in which we introduce an α value to regulate the weight of the KL divergence loss and the cross-entropy loss. We set α to 0 to exclusively use the cross-entropy part in the teacher model training. In contrast, during student model training, we set α to 0.5 to consider both parts of the loss with soft and hard pseudo-labels. It would be interesting as future work to experimentally search for an optimal value of α to explore its impact on performance.

To add the noise injection part, we utilize the pre-trained EN-DE/DE-EN and EN-RU/RU-EN back-translation models (Ng et al., 2019). Considering that transformer-based augmentation models can exponentially increase the computation time, we limit the amount of data to be augmented at 40% by setting the augmentation ratio to 0.4.

4.3 Experiment Results

The experimental results are presented in Table 2, where we evaluate the performance of our proposed approach, by measuring the accuracy on the validation dataset and comparing it with the baseline approaches. To analyze the results, we aim to answer the following research questions:

RQ1: Does domain shift exist in our task?

According to the first row of Table 2, training the model on product reviews and using it to predict peer reviews leads to very poor results. The accuracy scores are mostly around 50%, and some are even worse than random guessing. This suggests that the domain shift does exist in our case, and without applying any domain adaptation techniques, the model's performance will be poor.

RQ2: Is the performance different for different categories of product reviews?

In addition to assessing the existence of domain shift in our task, we are also interested in investigating the extent to which domain shift differed across various categories of product review data. Table 2 shows that the category "software" product review, which is more relevant to the peer review domain, yields better results than the "automotive" review. For example, when using the same 2000 labeled data, training on the “software” category yields 55.1% accuracy with the supervised learning baseline, while only 43.83% accuracy is achieved on the “automotive” category. After applying our proposed approach, we achieve 68.52% accuracy on “software” over 48.80% on “automotive” data. Hence, we conclude that source domain data with different relevance to the peer review data will result in varying degrees of the distributional gap, which is a crucial factor in domain-adaptation tasks.

RQ3: Does self-training mitigate domain shift by leveraging unlabeled data from the target

	Amazon "Software" data		*Amazon "Automotive" data*	
	500 labeled data	*2000 labeled data*	*500 labeled data*	*2000 labeled data*
Supervised Learning	6034%	55.1%	41.02%	43.83%
Self-training	60.67%	66.64%	42.31%	43.3%
Our Approach	**63.05%**	**68.52%**	**52.54%**	**48.61%**

Table 2: Accuracy scores of the proposed approach on various source domain labeled datasets

domain?

Examining the second row of Table 2, fairly good improvements can be observed by applying self-training. In addition to an average improvement of 3.16% in accuracy across all datasets, the greatest improvement of 11.54% is achieved with 2000 labeled "software" reviews. This convincingly demonstrates the benefits of learning from unlabeled target-domain data, even in the absence of labeled information. The results indicate considerable effectiveness of using self-training to tackle domain shift issues.

RQ4: Is our proposed approach able to enhance the performance of self-training?

We aim to assess whether our proposed approach is able to improve performance and overcome the limitations of the self-training baseline. The third row of Table 2 shows that our approach, which incorporates knowledge distillation and noise injection, outperforms the self-training baseline. We achieved the best accuracy score of 68.52%, the greatest improvement of 10.42%, and an average improvement of 4.95% over the self-training baseline. These results demonstrate that by incorporating knowledge distillation and noise injection, the student model learns more effectively and outperforms the teacher model.

RQ5: Does the effectiveness of the proposed approach depend on the size of the source-domain labeled dataset?

We perform experiments using different sizes of labeled datasets from the source domain. As presented in Table 2, the "software" dataset shows better performance with 2000 labeled reviews compared to 500 labeled reviews. Surprisingly, we find that for the "automotive" reviews, training with only 500 labeled reviews outperforms even 2000 labeled reviews. We hypothesize that with a less relevant source domain dataset, a larger labeled dataset can result in more misleading training due to a larger distributional gap. Furthermore, our proposed approach shows a greater improvement over the self-training baseline with 500 labeled reviews than with 2000 labeled reviews of both categories. This indicates that our approach is more effective in improving self-training, given that only limited data can be gleaned from the source domain.

	"Software" labeled data	*"Automotive" labeled data*
Self-training + kd	67.14%	44.87%
Self-training + noise	**68.9%**	43.03%
Our proposed approach	68.52%	**48.61%**

Table 3: Comparison of the accuracy scores by applying KD and noise injection respectively with self-training.

4.4 Ablation Study

In addition to the results presented in Table 2, we also examine the effect of each individual component in our proposed approach on the overall performance. We conduct extensive experiments by using only knowledge distillation or noise injection. The results are evaluated with the 2000 labeled reviews from both categories, which are shown in Table 3.

The table reveals some intriguing findings. We unexpectedly achieve a better result than our proposed approach by using only the noise injection, trained on the "software" labeled data. This indicates that using both components together may cause a performance drop. Similarly, we observe that using KD alone leads to better performance compared to noise injection alone, for the "automotive" review dataset. This contrasts with our finding for the "software" data. In the future, we plan to explore ways to optimize the use of both components and make them mutually beneficial.

5 Conclusion

This study first highlights the pedagogical significance of predicting helpful reviews in peer assessment to benefit student learning, and then considers the challenge of collecting labeled data to build a reliable prediction model. We explore a solution via domain adaptation to reduce the need of collecting labeled data. Our primary contribution is proposing self-training as an optimal domain-adaptation technique to address the domain-shift issue that commonly arises when transferring knowledge between domains. Furthermore, we incorporate knowledge distillation and noise injection into self-training to improve performance. The experimental results exhibit promise in utilizing self-training and show the effectiveness of our proposed approach. In addition, we discuss future work in optimizing the integration of knowledge distillation and noise injection.

References

Shai Ben-David and Ruth Urner. 2012. On the hardness of domain adaptation and the utility of unlabeled target samples. volume 7568 LNAI, pages 139–153.

Olivier Chapelle, Bernhard Scholkopf, and Alexander Zien. 2009. Semi-supervised learning. *IEEE Transactions on Neural Networks*, 20:542.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Hao Feng, Minghao Chen, Jinming Hu, Dong Shen, Haifeng Liu, and Deng Cai. 2021. Complementary pseudo labels for unsupervised domain adaptation on person re-identification. In *IEEE Transactions on Image Processing*, volume 30, pages 2898–2907.

Dilrukshi Gamage, Thomas Staubitz, and Mark Whiting. 2021. Peer assessment in moocs: Systematic literature review. *Distance Education*, 42:268–289.

Yaroslav Ganin and Victor Lempitsky. 2015. Unsupervised domain adaptation by backpropagation. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, page 1180–1189. JMLR.org.

Edward Gehringer, Luke Ehresman, Susan G GConger, and Prasad Wagle. 2006. Reusable learning objects through peer review: The expertiza approach. volume 3, pages 1–2.

Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Domain adaptation for large-scale sentiment classification: A deep learning approach. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ICML'11, page 513–520, Madison, WI, USA. Omnipress.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc.

Junxian He, Jiatao Gu, Jiajun Shen, and Marc'Aurelio Ranzato. 2020. Revisiting self-training for neural sequence generation. In *International Conference on Learning Representations*.

Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. 2015. Distilling the knowledge in a neural network.

Stella Ho, Ming Liu, Lan Du, Yunfeng Li, Longxiang Gao, and Shang Gao. 2022. Semi-supervised continual learning with meta self-training. In *Proceedings* *of the 31st ACM International Conference on Information amp; Knowledge Management*, CIKM '22, page 4024–4028, New York, NY, USA. Association for Computing Machinery.

Dong-Hyun Lee. 2013. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. pages 1–6.

Bo Li, Yezhen Wang, Tong Che, Shanghang Zhang, Sicheng Zhao, Pengfei Xu, Wei Zhou, Yoshua Bengio, and Kurt Keutzer. 2020. Rethinking distributional matching based domain adaptation. *arXiv preprint arXiv:2006.13352*.

Hongli Li, Yao Xiong, Charles Vincent Hunter, Xiuyan Guo, and Rurik Tywoniw. 2019. Does peer assessment promote student learning? a meta-analysis. *Assessment and Evaluation in Higher Education*, 45:1–19.

Hong Liu, Jianmin Wang, and Mingsheng Long. 2021. Cycle self-training for domain adaptation.

Mingsheng Long, Yue Cao, Jianmin Wang, and Michael I. Jordan. 2015. Learning transferable features with deep adaptation networks. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, page 97–105. JMLR.org.

Kristi Lundstrom and Wendy Baker. 2009. To give is better than to receive: The benefits of peer review to the reviewer's own writing. *Journal of Second Language Writing*, 18:30–43.

Ke Mei, Chuang Zhu, Jiaqi Zou, and Shanghang Zhang. 2020. Instance adaptive self-training for unsupervised domain adaptation. In *European Conference on Computer Vision (ECCV)*.

Melissa M. Nelson and Christian D. Schunn. 2009. The nature of feedback: How different types of peer feedback affect writing performance. *Instructional Science*, 37:375–401.

Nathan Ng, Kyra Yee, Alexei Baevski, Myle Ott, Michael Auli, and Sergey Edunov. 2019. Facebook fair's wmt19 news translation task submission. In *Proc. of WMT*.

Jianmo Ni, Jiacheng Li, and Julian McAuley. 2019. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 188–197, Hong Kong, China. Association for Computational Linguistics.

Xiaokun Pu and Chunguang Li. 2023. Meta-self-training based on teacher–student network for industrial label-noise fault diagnosis. *IEEE Transactions on Instrumentation and Measurement*, 72:1–11.

Xianshan Qu, Xiaopeng Li, Csilla Farkas, and John Rose. 2020. An attention model of customer expectation to improve review helpfulness prediction. In *Advances in Information Retrieval*, pages 836–851, Cham. Springer International Publishing.

Lakshmi Ramachandran, Edward F. Gehringer, and Ravi K. Yadav. 2017. Automated assessment of the quality of peer reviews using natural language processing techniques. *International Journal of Artificial Intelligence in Education*, 27:534–581.

Keith J. Topping. 2009. Peer assessment. *Theory into Practice*, 48:20–27.

Oren Tsur and Ari Rappoport. 2009. Revrank: A fully unsupervised algorithm for selecting the most helpful book reviews. *Proceedings of the International AAAI Conference on Web and Social Media*, 3(1):154–161.

Qian Wang and Toby Breckon. 2020. Unsupervised domain adaptation via structured prediction based selective pseudo-labeling. volume 34, pages 6243–6250.

Wikipedia contributors. 2023. Kullback–leibler divergence — Wikipedia, the free encyclopedia.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Yunkai Xiao, Tianle Wang, Xinhua Sun, Yicong Li, Yang Song, Jialin Cui, Qinjin Jia, Chengyuan Liu, and Edward F. Gehringer. 2022. Modeling review helpfulness with augmented transformer neural networks. In *2022 IEEE 16th International Conference on Semantic Computing (ICSC)*, pages 83–90.

Yunkai Xiao, Gabriel Zingle, Qinjin Jia, Shoaib Akbar, Yang Song, Muyao Dong, Li Qi, and Edward Gehringer. Problem detection in peer assessments between subjects by effective transfer learning and active learning. In *Proceedings of the 13th International Conference on Educational Data Mining (EDM 2020)*.

Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. 2020. Unsupervised data augmentation for consistency training. volume 33, pages 6256–6268. Curran Associates, Inc.

Wenting Xiong and Diane Litman. 2011a. Automatically predicting peer-review helpfulness. pages 502–507. Association for Computational Linguistics.

Wenting Xiong and Diane Litman. 2011b. Understanding differences in perceived peer-review helpfulness using natural language processing. In *Proceedings of the Sixth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 10–19.

Yinfei Yang, Yaowei Yan, Minghui Qiu, and Forrest Bao. 2015. Semantic analysis and helpfulness prediction of text for online product reviews. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 38–44.

Fei Yu, Mo Zhang, Hexin Dong, Sheng Hu, Bin Dong, and Li Zhang. 2021. Dast: Unsupervised domain adaptation in semantic segmentation based on discriminator attention and self-training. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 10754–10762.

Gabriel Zingle, Balaji Radhakrishnan, Yunkai Xiao, Edward Gehringer, Zhongcan Xiao, Ferry Pramudianto, Gauraang Khurana, and Ayush Arnav. 2019. Detecting suggestions in peer assessments. In *Proceedings of the 12th International Conference on Educational Data Mining (EDM 2019)*, pages 474–479. International Educational Data Mining Society.

Yang Zou, Zhiding Yu, Xiaofeng Liu, B. V.K.Vijaya Kumar, and Jinsong Wang. 2019. Confidence regularized self-training. volume 2019-Octob, pages 5981–5990.

Yasemin Demiraslan Çevik, Tülin Haşlaman, and Serkan Çelik. 2015. The effect of peer assessment on problem solving skills of prospective teachers supported by online learning activities. *Studies in Educational Evaluation*, 44:23–35.