Journal of Harbin Engineering University
ISSN: 1006-7043

Vol 44 No. 12
December 2023

Cross-Domain Aspect Extraction using Adversarial Domain Adaptation

[1]Sampathirao Suneetha, [2] Prof. S. Vizianandа Row

[1]Research Scholar, [2]Professor
[1,2]Department of CS&SE, AUCE, Andhra University, Visakhapatnam, AP, India.
[1]Corresponding author email: sampath.suneetha@gmail.com

ABSTRACT: Aspect extraction, the task of identifying and categorizing aspects or features in text, plays a crucial role in sentiment analysis. However, aspect extraction models often struggle to generalize well across different domains due to domain-specific language patterns and variations. In order to tackle this challenge, we propose an approach called "Cross-Domain Aspect Extraction using Adversarial-Based Domain Adaptation". Our model combines the power of pre-trained language models, such as BERT, with adversarial training techniques to enable effective aspect extraction in diverse domains. The model learns to extract domain-invariant aspects by incorporating a domain discriminator, making it adaptable to different domains. We evaluate our model on datasets from multiple domains and demonstrate its effectiveness in achieving cross-domain aspect extraction. The results of our experiments reveal that our model outperforms baseline techniques, resulting in significant gains in aspect extraction across various domains. Our approach opens new possibilities for domain adaptation in aspect extraction tasks, providing valuable insights for sentiment analysis in diverse domains.

Keywords: Aspect Extraction, Adversarial Domain Adaption, Aspect Based Sentiment Analysis, Domain Adaption, Cross Domain.

1. INTRODUCTION

Cross-domain aspect extraction refers to extracting fine-grained aspects or sub-topics from textual data in source domain and applying the knowledge to a target domain. It plays a crucial role in various NLP applications, such as sentiment analysis, review summarization, and opinion mining. However, the main challenge in cross-domain aspect extraction lies in the distributional differences between the source domain, which has labeled data, and the target domain, where the model must perform extraction without the availability of labeled data [1].

To tackle this challenge, adversarial discriminative domain adaptation techniques have emerged as effective solutions. The objective of these techniques is to establish alignment between the feature representations of the source and target domains [2]. This alignment enables the model to generalize effectively to the target domain, even when labeled data specific to the target domain is unavailable. One popular model for performing aspect extraction is the BERT model. BERT has demonstrated remarkable performance in various NLP tasks due to its ability to capture contextual information and semantic relationships within the text.

Our study proposes a cross-domain aspect extraction method that leverages adversarial discriminative domain adaptation and the BERT model. We aim to learn domain-invariant representations as part of the aspect extraction process so that the model can generalize effectively to the target domain. Specifically, we employ a domain adaptation framework comprising three key components: a feature extractor based on the BERT model, a domain classifier, and an aspect extractor.

The feature extractor encodes input sentences into contextualized word embeddings using pre-trained BERT. These embeddings capture the rich semantic and contextual information necessary for aspect extraction. The domain classifier distinguishes between the source domain and target domain by learning domain-specific features. Simultaneously, the aspect extractor utilizes the encoded representations to identify the fine-grained aspects within the text.

We introduce an adversarial training mechanism between the aspect extractor and the domain classifier to encourage domain-invariant feature learning. The domain classifier attempts to accurately classify the extracted aspect domain, while the aspect extractor aims to generate domain-invariant representations that confuse the classifier. This adversarial training process promotes the alignment of feature distributions between the two domains.

By incorporating this adversarial discriminative domain adaptation approach with the BERT model, we aim to enhance the performance of cross-domain aspect extraction. To evaluate the effectiveness of our proposed method in adapting the aspect extraction model from the source to the target domain, we conducted experiments using benchmark datasets from various domains. The results of these experiments demonstrate the efficacy of our approach. The results show significant

improvements in aspect extraction accuracy, highlighting the potential of our approach in addressing the domain shift problem in cross-domain aspect extraction.
In conclusion, our work contributes to cross-domain aspect extraction by leveraging adversarial discriminative domain adaptation techniques and the powerful BERT model. By aligning feature distributions between different domains, we enable the BERT model to generalize effectively to new domains without requiring labeled target domain data. Our proposed method demonstrates promising results and opens avenues for further research in adapting NLP models across different domains.

2. RELATED WORK

Sentiment analysis encompasses the methodical process of identifying, extracting, quantifying, and analyzing emotional states and subjective information through the integration of text analysis, natural language processing, bio-metrics, and computational linguistics [3]. This approach is widely employed for the analysis of customer feedback materials such as reviews and survey responses, online and social media content, as well as healthcare-related materials. Its applications span across various domains, including marketing, customer service, and clinical medicine.
Aspect-based sentiment analysis serves as a specialized sub-field within sentiment analysis, concentrating on the identification of sentiments expressed towards aspects or features of a service or product [4]. It provides more detailed and actionable insights than traditional sentiment analysis, providing only an overall sentiment score. Several methods utilizing deep learning techniques have been proposed to address the ABSA problem [5], [6],[7].
In one study [8], a CNN architecture with seven-layers is designed, and both word embeddings and POS tagging are used as features. In an alternative approach [9], convolutional neural networks are employed alongside domain-specific data for aspect extraction and aspect sentiment classification. It demonstrates that incorporating domain-specific word embeddings enhances the semantic enrichment of general-purpose embeddings for the given task. Recent research [10] also shows that utilizing in-domain data improves the performance of state-of-the-art language models such as BERT. In a similar vein, [10] employs a two-stage process for aspect-based sentiment classification (ASC), where BERT is fine-tuned on domain-specific data. This process includes two stages: supervised task-specific fine-tuning and self-supervised in-domain fine-tuning. By following this approach, we achieve enhanced performance for ASC. Another recent study [12] introduces a new taxonomy for ABSA, organizing existing studies based on sentiment elements and emphasizing compound ABSA tasks. Furthermore, in [13], a comprehensive solution for ABSA is introduced. This solution involves the joint training of two BERT-MRC (machine reading comprehension) models with shared parameters and the construction of two MRC problems.
Aspect extraction is a crucial step in opinion mining, as effective extraction leads to a powerful opinion mining system [14]. Previous approaches have used methods such as association rule mining [15] and an unsupervised framework known as OPINE [16] for aspect extraction. Several research papers have explored aspect extraction using BERT. For example, [17] presents a method that utilizes aspects extracted through unsupervised techniques as labels for training a hierarchical attention-based network, leveraging a pre-trained BERT2 language model.
Developing robust solutions that can effectively generalize to unseen and out-of-domain examples poses a fundamental challenge. It necessitates the design of models that can effectively adapt and perform well in scenarios beyond the training data, enabling reliable performance in real-world applications. Previous domain adaptation approaches focused on learning domain-invariant latent features, aiming to minimize the distance between features from the source and target domains [18, 19]. Deep Neural Networks (DNNs) have introduced a shift towards unified approaches, where domain-invariant feature transformations and task-specific classifiers are jointly learned during training, leading to monolithic models that encompass both aspects. These methodologies incorporate advanced techniques such as the Gradient Reversal Layer [20] and DNN explicit partitioning [21] to enable the implicit learning of both domain-specific and domain-invariant features in an end-to-end fashion. They have found application in various NLP tasks, including cross-domain sentiment analysis. [22,23] Introduce additional training tasks for BERT [24] to acquire both domain-specific and domain-invariant feature representations for sentiment analysis tasks. Moreover, leveraging syntactic information has proven effective in introducing domain-invariant knowledge to bridge the gap between different domains [25,26].
Cross-domain aspect extraction refers to identifying aspects in text data from a different domain than the one used for training the model.

This can be challenging due to differences in language use and aspect relevance between domains. Cross-Domain Aspect Extraction has also been explored in several works. In [27], a novel approach is presented, which automatically constructs domain-specific knowledge graphs encompassing pertinent information for aspect term identification. [28] centers on Cross-Domain Aspect Extraction and explores diverse language models and embedding techniques.
A label propagation algorithm is presented in [29] to extract aspects from heterogeneous networks. This algorithm leverages linguistic features to facilitate the propagation process. It involves a transductive learning approach that integrates both labeled and unlabeled aspects in the label propagation step. Adversarial training has been commonly employed for sentence classification tasks, but its application and effects in aspect-based sentiment analysis (ABSA) have not been extensively investigated. Therefore, this study aims to examine the influence of employing adversarial training on the BERT language model, a highly effective model, for extracting aspects across different domains.

3. PROBLEM STATEMENT

The problem involves cross-domain aspect extraction, focusing on the source domain (D_s) and the target domain (D_t). In the source domain, we have a set of labeled examples, denoted as $\{x_i^s, y_i^s\}_{i=1}^{N_s}$, where x_i^s represents a sentence and y_i^s is the corresponding aspect label. In the target domain, we have a collection of unlabeled data (D_t), denoted as $\{x_i^t\}_{i=1}^{N_t}$, where N_t represents the number of unlabeled sentences. The objective is to develop a robust classifier trained on labeled data from the source domain, capable of extracting aspect terms from unlabeled sentences in the target domain.

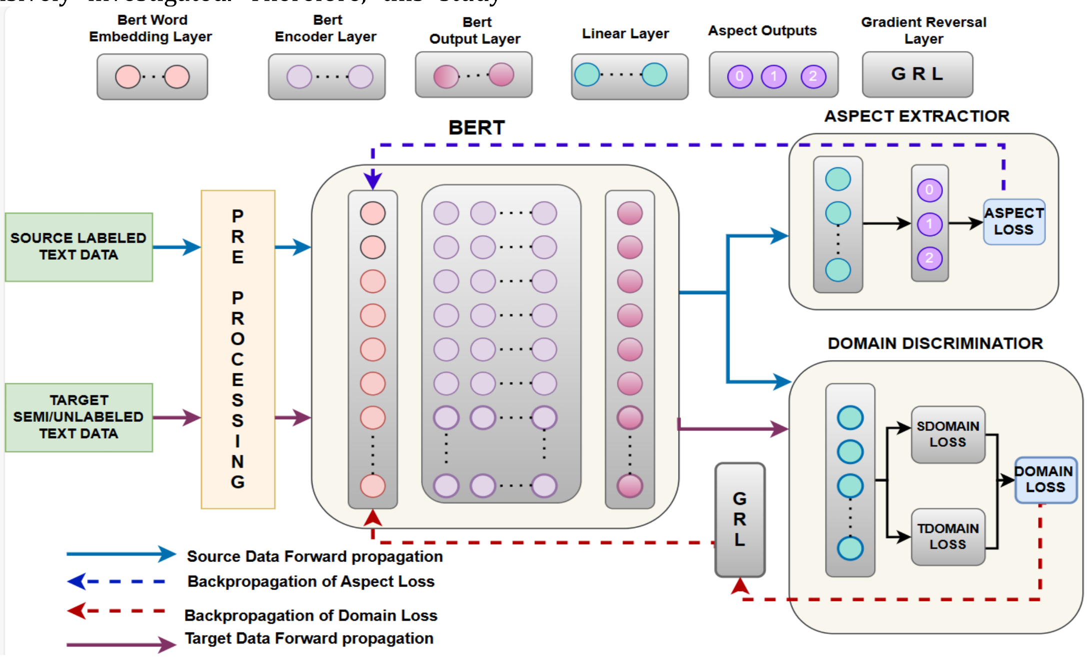


Figure 1: The proposed architecture: Cross Domain Aspect Extraction using Adversarial Domain Adaptation (CDAE-ADA)

4. METHODOLOGY

This methodology can be theoretically justified by the following two principles:
Domain invariance: The goal of domain invariance is to ensure that the model learns a representation of the text reviews independent of the domain. This is significant since it enables the model to be transferred to other domains without retraining.
Aspect specificity: Aspect specificity aims to ensure that the model learns a representation of the text reviews that can distinguish between different aspects. This is crucial as it enables the model to conduct sentiment analysis at the aspect level.
Our approach comprises a four-step process: pre-processing, transfer learning, multi-task learning, and adversarial training. In the following sections, we will provide a comprehensive overview of the methodology for each step, as depicted in Figure 1.

4.1 Pre-processing: In this step, we pre-process the input data to prepare it for training. The pre-processing involves several tasks: We begin by converting the text data into lowercase to ensure consistency in the representations. Additionally, we remove punctuation marks such as periods, commas, and quotation marks from the text to eliminate their influence on the learning process. Next, we tokenize the text, splitting it into individual tokens or words. This allows us to capture the finer linguistic details and enables the model to process the text at the token level.

We need to convert the text labels into aspect tags to train the aspect extraction component. We assign the following labels to the tokens: 0 for unrelated, 1 for the start of an aspect term, and 2 for the middle or end of an aspect term. For example, a sentence like "The battery life is excellent" might be labeled as "The [0] battery [1] life [2] is [0] excellent [0].

After converting the text into a numerical format, we create a vocabulary of unique tokens and assign a unique index to each token. Each token in the text is replaced with its corresponding index to create a numerical representation of the text.

We pad the sequences to ensure that all input sentences have the same length. Sentences shorter than the maximum sequence length is padded with special padding tokens, while longer sentences are truncated. This step enables efficient batch processing during training and ensures consistent input dimensions for the model.

Finally, we create data loaders to handle the loading and pre-processing of data in batches during training. Data loaders improve training speed and memory utilization by efficiently managing the batch processing of the data.

4.2 Transfer Learning: It involves leveraging pre-trained models to benefit a target task. In our work, the BERT is trained on large-scale datasets to learn general language patterns. During transfer learning, the BERT model's parameters are partially updated, and two additional task-specific layers are added on top of the BERT output layers for aspect extraction and domain classification. Using a smaller task-specific data-set, the model is subsequently fine-tuned for a specific target task. The idea is that the pre-trained model has already learned useful representations, and fine-tuning allows the model to adapt to the specific task [30].

4.2.1 BERT Word Embedding Layer: BERT employs a word embedding layer that transforms input tokens into dense vector representations capturing semantic and contextual information. Unlike traditional word embeddings, BERT uses contextual word embeddings based on Word-pieces, sub word units created through tokenization. These Word-pieces allow BERT to handle out-of-vocabulary words and capture morphological variations. The word embedding layer in BERT consists of token embeddings representing individual Word-pieces, segment embeddings distinguishing between different sentences, and position embeddings encoding token positions. Together, these embeddings form the input representation for BERT, providing contextual and positional information [31]. During fine-tuning, the word embedding layer and the rest of the model are updated to adapt the embeddings to the specific task, refining the model's understanding of word meanings and contextual relationships based on the labeled data.

4.2.2 BERT Encoder: The BERT Encoder is a key component of the BERT model used in transfer learning. It consists of a stack of transformer layers that process the input token embeddings in a self-attention mechanism. Each transformer layer in the BERT Encoder consists of multiple attention heads that capture different aspects of the input's contextual information. Within each attention head, self-attention is performed to compute attention weights for each token, allowing the model to focus on relevant tokens while encoding contextual relationships. The attention weights are then applied to compute weighted sums of the token embeddings, capturing the contextualized representations. Additionally, each transformer layer includes feed-forward neural networks that apply non-linear transformations to the contextualized representations. The BERT Encoder's multi-layered structure and self-attention mechanism enable it to capture rich contextual information from the input tokens, facilitating better understanding and representation of language patterns [32].

4.2.3 Linear Layer: In the given program, a fully connected layer/linear layer is used for both the aspect extraction and domain classification tasks. After the feature extraction using the BERT model, separate linear layers are added on top to perform these tasks. The fully connected layers consist of a set of trainable weights and biases, which are applied to the input features to produce the desired output [33]. To address the aspect extraction task, the output of the fully connected layer is subjected to a softmax activation function, yielding probabilities for each aspect category. This facilitates the model in predicting the likelihood of the input sentence belonging to different aspect classes. Similarly, for the domain classification task, the output of the fully connected layer undergoes a softmax activation function, resulting in probabilities assigned to each domain class. This enables the model to estimate the probability of the input sentence pertaining to various domain classes.

Loss Function: The loss function used for both tasks is the categorical cross-entropy loss. It compares the predicted probabilities with the true labels for each task and calculates the loss based on the prediction error [34]. The categorical cross-entropy loss function is widely employed in multi-class classification tasks.

Using separate fully connected layers and the appropriate loss function for each task, the model can simultaneously learn task-specific representations and optimize its performance on aspect extraction and domain classification tasks.

4.3 Multi-Task Learning: It involves simultaneously training a model on multiple related tasks. Our approach employs multi-task learning, training a model simultaneously on aspect extraction and domain classification tasks. With a pre-trained BERT model as the backbone, we extract features from the input text $(\mathbf{x_s}, \mathbf{x_t})$ using the feature extractor $\mathbf{f_\theta}$. This results in feature vectors $\mathbf{z_s} = \mathbf{f_\theta(x_s)}$, $\mathbf{z_t} = \mathbf{f_\theta(x_t)}$, then we add specific layers for each task and train them jointly. By leveraging shared representations, the model captures both aspect and domain information, optimizing performance on both tasks. During training, the aspect extractor $\mathbf{f_s}$ takes the feature vector $\boldsymbol{z_s}$ as input and outputs a probability distribution over the aspect labels. This probability distribution can be used to predict the aspect labels for the input text. Then the aspect loss $\mathbf{L_{as}}$ is calculated by comparing the predicted aspect labels with the ground-truth labels using the cross-entropy loss function. The aspect loss can be calculated as follows [34]:

$$\mathbf{L_{as}} = -\sum_{i=1}^{n} \mathbf{y_s^i} \mathbf{log}(\mathbf{f_s}(\mathbf{X_s^i}))$$

Where n is the number of examples in the source data, $\mathbf{y_s^i}$ is the ground-truth aspect tag for the i^th^ example, and $\mathbf{f_s}(\mathbf{X_s^i})$ is the predicted aspect tag for the i^th^ example. Next, the computation of the gradients of the loss function with respect to the model parameters can be expressed as follows:

$$\nabla\theta\mathbf{L_{as}} = \frac{\partial \mathbf{L_{as}}}{\partial \theta_{as}}$$

Then backpropagate the aspect loss $\mathbf{L_{as}}$ and update the parameters of $\mathbf{f_s}$. Back-propagate the gradients through the layers of the model to update the model parameters($\boldsymbol{\theta}$):

$$\boldsymbol{\theta}\_\mathbf{new} = \boldsymbol{\theta} - \boldsymbol{\alpha} * \nabla\theta\mathbf{L_{as}}$$

Here, α represents the learning rate, which determines the step size for parameter updates during optimization. The domain classifier $\mathbf{g_s}$ takes the feature vector $\boldsymbol{z_s}$ as input and outputs a probability distribution over the domain labels. This probability distribution can be used to predict the domain of the source input text. Then the domain loss $\mathbf{L_{dom_s}}$ is calculated by comparing the predicted domain labels with the ground-truth labels using the cross-entropy loss function. Similarly, the domain classifier $\mathbf{g_t}$ takes the feature vector $\mathbf{z_t}$ as input and outputs a probability distribution over the domain labels. This probability distribution can be used to predict the domain of the source input text. Then The domain loss $\mathbf{L_{dom_t}}$ is calculated by comparing the predicted domain labels with the ground-truth labels using the cross-entropy loss function. The domain loss can be calculated as follows:

$$\mathbf{L_{dom_s}} = -\sum_{i=1}^{n} \mathbf{D_s^i} \mathbf{log}(\mathbf{g_s}(\mathbf{X_s^i}))$$
$$\mathbf{L_{dom_t}} = -\sum_{i=1}^{m} \mathbf{D_t^i} \mathbf{log}(\mathbf{g_t}(\mathbf{X_t^i}))$$

Where m is the number of examples in the target data, $\mathbf{D_s^i}$ is the ground-truth domain class for the i^th^ instance in the source data, and $\mathbf{D_t^i}$ is the ground-truth domain class for the i^th^ example in the target data.

The domain loss $\mathbf{L_{dom}}$ can be calculated as follows:

$$\mathbf{L_{dom}} = \mathbf{L_{dom_s}} + \mathbf{L_{dom_t}}$$

The gradients of the loss for the model parameters:

$$\nabla\theta\mathbf{L_{dom}} = -\lambda \frac{\partial \mathbf{L_{dom}}}{\partial \theta_{dom}}$$

Where $\boldsymbol{\lambda}$ is the gradient reversal coefficient. Back-propagate the gradients through the layers of the model to update the parameters:

$$\boldsymbol{\theta}\_\mathbf{new} = \boldsymbol{\theta} - \boldsymbol{\alpha} * \nabla\theta\mathbf{L_{dom}}$$

This multi-task learning framework allows the model to benefit from shared knowledge, enhancing aspect extraction even without labeled data in the target data-set.

4.4 Adversarial Domain Adaptation: In our proposed approach, adversarial training is employed to address the cross-domain aspect extraction problem.

The model's architecture comprises two primary components shown in Figure 2, the aspect extractor, serving as the generator, and the domain discriminator, acting as the discriminator.

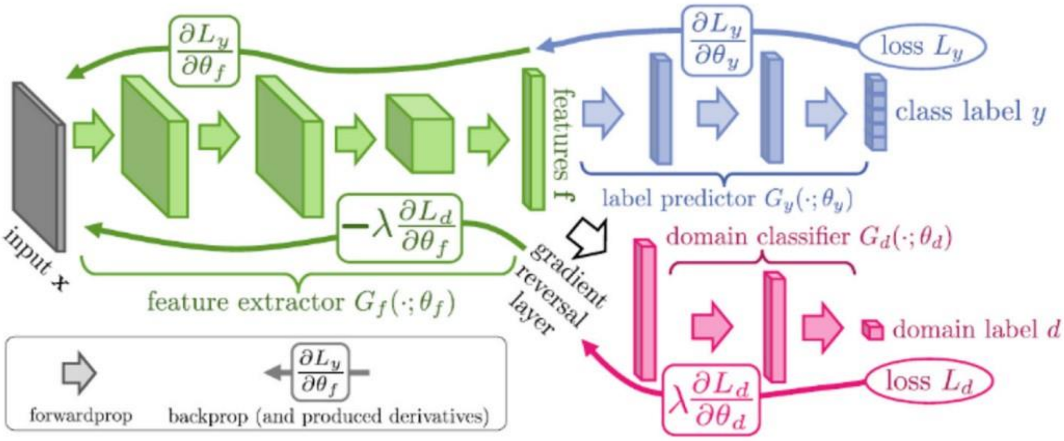


Figure 1 : Domain-adversarial neural network

Figure 2 architecture [28].

The aspect extractor, functioning as the generator, aims to generate domain-invariant representations of domain-specific information. Its objective is to accurately extract aspect terms from the input data. On the other hand, the domain discriminator's role is to classify the domains correctly based on the input representations.

During training, the aspect extractor extracts aspect terms from feature extractor's representations of the source dataset and calculates the aspect extraction loss by comparing the generated aspect outputs with the labeled data. The aspect extractor is updated by minimizing this loss to identify aspect terms accurately.

Simultaneously, the feature extractor's representations of both source and target data are passed through the domain discriminator, which predicts the domain class. The domain discriminator calculates the domain loss. To align the objectives of the aspect extraction and domain adaptation tasks, a gradient reversal layer having a gradient reversal function, which is applied when back-propagating the domain loss $\mathbf{L_{dom}}$ with coefficient λ. This reversal of gradients ensures that the aspect extractor maximizes the domain loss, promoting the generation of domain-invariant representations. The gradient reversal function can be applied to $\mathbf{L_{dom}}$ as follows:

$$\mathbf{L_{dom}^{rev}} = -\boldsymbol{\lambda}\mathbf{L_{dom}}$$

where $\boldsymbol{\lambda}$ is the gradient reversal coefficient, and $\mathbf{L_{dom}^{rev}}$ is the modified domain loss after applying the gradient reversal function.

Through an iterative process, the aspect extractor and the domain discriminator engage in a competitive interplay. The aspect extractor strives to generate aspect terms that are challenging for the domain discriminator to classify accurately. In contrast, the domain discriminator adapts to more effectively distinguish the domains based on the representations. This adversarial training encourages the aspect extractor to learn domain-invariant features that are useful for the aspect extraction task in various domains.

In summary, the adversarial training framework facilitates the learning of domain-invariant representations by simultaneously minimizing the aspect loss to accurately extract aspect terms and maximizing the domain loss to ensure the generated representations are not specific to any domain. This approach enables the model to generate aspect terms with domain-invariant features, suitable for application in different domains, even without labeled data in the target domain.

The overall formula for the cross-domain aspect extraction using an adversarial domain adaptation approach can be defined as follows:

$$\mathbf{L = L_{as} + L_{dom}^{rev}}$$

Where L is the overall loss. This can be written like this also

$$\mathbf{L} = -\sum_{i=1}^{n} \mathbf{y_s^i \log(f_s(X_s^i))} + \boldsymbol{\lambda}\left(-\sum_{i=1}^{n} \mathbf{D_s^i \log(g_s(X_s^i))} - \sum_{i=1}^{m} \mathbf{D_t^i \log(g_t(X_t^i))}\right)$$

5. EXPERIMENTS

5.1 Datasets: To evaluate our approach's cross-domain aspect extraction performance, we utilize two benchmark ABSA datasets containing English-language consumer reviews for restaurants and laptops. These two datasets are mostly used in ABSA research because of the limited availability of labeled data for aspect labeling. The task of aspect labeling is challenging and time-consuming, which makes it difficult to obtain a large amount of annotated data. To assess cross-domain performance, we create pairings of the two data domains: L and R represent the laptops and restaurants review datasets, respectively.

Within each domain, we partition the data into two separate subsets, which are then divided into train and test sets using 8:2 ratio. A summary of these datasets can be seen in Table I.

Data-set	Description	Total	Train	Test
R	Restaurants	5,841	4673	1168
L	Laptops	3,845	2364	590

Table 1 : Restaurant and Laptop datasets from SemEval 2014.

5.2 Implementation details: In our experiments, we adopt the Transformer model implementations provided by Hugging Face [35]. Specifically, we utilize the code base of BERT and fine-tune the model using the Adam optimizer [36] with a learning rate of 2e−5. For the CDAE-ADA model experiments, we set the batch size to 16. We performed our experiments on a system equipped with a Windows operating system with, an 11th Gen Intel Core i5 CPU, and 8GB of RAM and GPUP100. During the code execution, we encountered an issue with the input data. We have many unique aspect terms in a data-set and one sentence may have more than one aspect terms. To address these, we converted the text tokens into tags, as discussed in section 4.1.

5.3 Hyper-parameters and adversarial training: To conduct our experiments, we initialize our model with BERT, utilizing of BERT-BASE-uncased version on restaurant and laptop data. We performed several experiments to ascertain the optimal dropout probability and number of training epochs specifically for BERT. The results for 1 to 6 training epochs are depicted. For AE (aspect extraction), we experimented with five different dropout values (0.1, 0.2, 0.3, 0.4, 0.5) in the linear layer. The results for cross-domain restaurant and laptop datasets can be seen in Figure 3. In Figure 3, we present two subplots. Subplot (a) illustrates the F1 Score plotted against the number of epochs for different dropout values in Restaurant to Laptop scenario. Subplot (b) showcases the F1 Score plotted against the number of epochs for different dropout values in Laptop to Restaurant scenario. Based on these results, we infer that the test set examples from the laptop domain exhibit more similar patterns than those from the restaurant domain. This observation is supported by the relatively higher F1 scores achieved when training on the restaurant domain and applying the model to the laptop domain, compared to the reverse scenario. The F1 scores for aspect extraction in the Laptop to Restaurant scenario vary between 0.33 and 0.68. These results suggest that the model, which was originally trained on the laptop domain, faces difficulties when it comes to generalizing effectively to the restaurant domain. This suggests that the patterns and characteristics of the restaurant domain examples differ from those of the laptop domain examples.

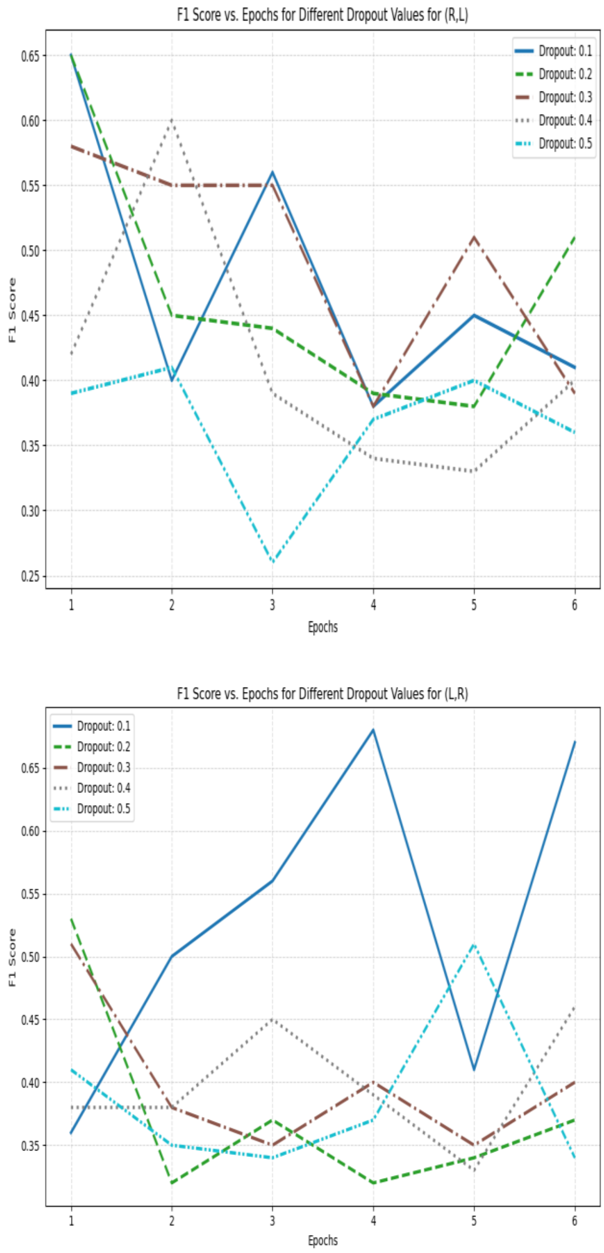


Figure 3 : (a) F1 Score vs. Epochs for Different Dropout Values for (R, L) and (b) F1 Score vs. Epochs for Different Dropout Values for (L, R)

Lambda value	(L, R)			(R, L)		
	Macro Average F1 Score	Weighted Average F1 score	Accuracy	Macro Average F1 Score	Weighted Average F1 score	Accuracy
0.01	68	92	93	65	91	92
0.1	66	89	91	63	88	90
0.5	54	80	81	55	82	83
1.0	28	53	54	26	51	52

On the other hand, in the Restaurant to Laptop scenario, the F1 scores show better performance, ranging from 0.26 to 0.65. Although the scores are not extremely high, they indicate that the model trained on the restaurant domain adapts more quickly to the laptop domain. This implies that the patterns and characteristics of the laptop domain examples are like those of the restaurant domain examples.

Table 2: Comparison of Weighted Average F1 Scores and Accuracy of Our Method with Different Lambda Values

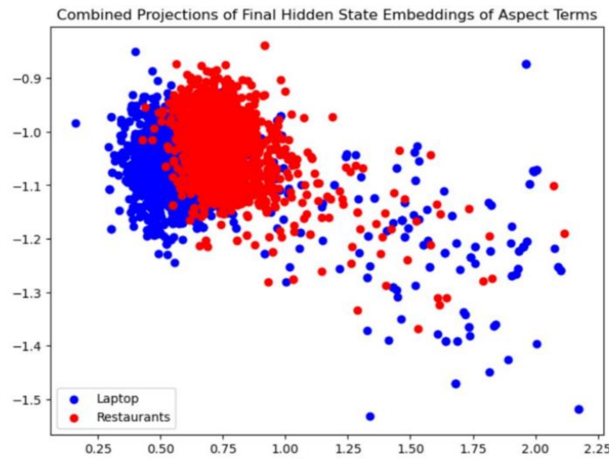


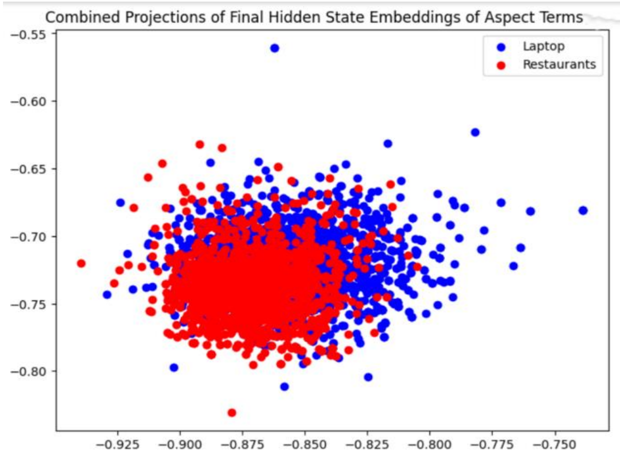
Figure 4 : Visualizing t-SNE Projections of Final Hidden State Embeddings of Aspect Terms Generated by CDAE-ADA: Laptop-to-Restaurant (a) and Restaurant-to-Laptop (b) Domain Adaptation, Represented by Purple (Laptops) and Red (Restaurants) Colors.

To illustrate the impact of adversarial domain adaption, Figures 4a and 4b depict t-SNE projections of the aspect terms' final hidden state embeddings produced by CDAE-ADA for the laptop to restaurant and restaurant to laptop domains, represented by purple (laptops) and red (restaurants) colors. These plots demonstrate how the two domains are invariant to each other.

5.4 Lambda value in adversarial training:

To evaluate the impact of adversarial domain adaptation on the model's performance, we will determine the optimal dropout value for each epoch count and conduct experiments using four distinct lambda (λ) values. The lambda value is a hyper-parameter used in the gradient reversal layer for domain adaptation. We evaluate different lambda values, including 0.01, 0.1, 0.5, and 1.0. The weighted avg F1 scores of aspect terms and accuracy for cross-domain laptop and restaurant datasets can be seen in Table-2.

These results illustrate the impact of different lambda values on adversarial domain adaptation. Lower lambda values (0.01, 0.1) generally yield better performance in terms of F1 scores and accuracies. As the lambda value increases (0.5, 1.0), the performance decreases, indicating a trade-off between domain adaptation and overall performance. The direction of domain adaptation also influences the results, with better performance observed when adapting from the Laptop to Restaurant domain compared to the reverse scenario. These findings highlight the importance of selecting an appropriate lambda value and understanding the domains' characteristics for effective domain adaptation.



Input Text	Predicted Aspects (Laptop to Restaurant)	Predicted Aspects (Restaurant to Laptop)
For the price you pay this product is very good. However, battery life is a little lack-luster coming from a MacBook Pro.	['price', 'battery life']	['price', 'battery life', 'mac']
Not was the food outstanding, but the little 'perks' were great.	['food', '##ks']	['food', 'per', '##ks']

The food is uniformly exceptional, with a very capable kitchen which will proudly whip up whatever you feel like eating, whether it's on the menu or not.	['food', 'kitchen']	['food', 'kitchen', 'menu']
I charge it at night and skip taking the cord with me because of the good battery life.	['cord', 'battery life']	['cord', 'battery life']

Table 3: Aspect Prediction Results

The table 3 provides an overview of the aspect prediction results for laptop-to-restaurant and restaurant-to-laptop adaptation scenarios. The predicted aspects shed light on the model's understanding of domain-specific aspects and its performance in adapting from one domain to another.

5.5 Results: We assess the performance of state-of-the-art Transformer models on cross-domain aspect extraction, incorporating adversarial domain adaptation as elaborated in Section-4. Our proposed model, CDAE-ADA, achieves the highest F1 scores compared to other existing solutions for cross-domain aspect extraction. This includes ARNN-GRU [37], which enhances an RNN architecture with GRU blocks by incorporating information from dependency trees through an auxiliary dependency relation classification task. SA-EXAL [38] is a BERT-like model that integrates syntactic information into its self-attention mechanism. TRNN-GRU [39] extends ARNN-GRU by including a conditional domain adversarial network to align word feature spaces between the source and target domains. PT-KI-BERT [40] presents a comprehensive approach involving the construction of domain-specific knowledge graphs and determining the benefits of injecting knowledge into Transformers for aspect extraction.

Model	(L, R)-F1 score	(R, L)-F1 score
ARNN-GRU	52.9	40.4
TRNN-GRU	53.8	40.2
SA-EXAL	54.7	47.6
PT-KI-BERT	66.4	49.9
BERT	45.1	44.6
DeBERTa	54.3	47.5
CDAE-ADA	68	65

Table 4: Comparative analysis of macro avg F1 scores for aspect extraction.

Furthermore, our experimental setup encompasses several baseline models to perform ablation studies. These models include BERT and DeBERTa, which were fine-tuned on the AE task without incorporating adversarial domain adaptation. Table 4 presents the macro avg of aspect extraction F1 scores in each cross-domain setting for our adversarial domain adaptation model and the baseline models. The results demonstrate that the CDAE-ADA model outperforms both the baseline models and models from other papers (ARNN-GRU, TRNN-GRU, SA-EXAL, PT-KI-BERT, BERT, and DeBERTa) in terms of F1 scores. The performance of the models varies across the two scenarios, with some models performing better in the (Laptop, Restaurant) scenario and others performing better in the (Restaurant, Laptop) scenario. Overall, the CDAE-ADA model demonstrates superior performance in both scenarios, indicating its effectiveness for the task of cross-domain aspect extraction.

6. CONCLUSION:

This work presented a novel approach called Cross-Domain Aspect Extraction using Adversarial-based Domain Adaptation (CDAE-ADA) for effective aspect extraction in ABSA. Our model combines the power of pre-trained language models, such as BERT, with adversarial training techniques to extract domain-invariant aspects, enabling adaptation to diverse domains.

Through extensive evaluations on labeled datasets from multiple domains, we demonstrated the superiority of our CDAE-ADA model over baseline models and existing approaches. Our model achieved the highest F1 scores, indicating its effectiveness in cross-domain aspect extraction. We also investigated the impact of hyperparameters and adversarial training, discovering that lower lambda values yielded better performance. This highlights the significance of selecting appropriate lambda values and gaining insights into domain characteristics.

Overall, our CDAE-ADA model showcased its potential for effective aspect extraction across different domains, making significant contributions to the field of ABSA. Our model opens up new possibilities for domain adaptation in aspect extraction tasks by addressing the challenge of domain-specific language patterns and variations.

While our CDAE-ADA model has demonstrated promising results in the context of cross-domain aspect extraction with two datasets, there is potential for expanding our approach to a multi-domain setting.

REFERENCES

1. Don Wisdom ," What Is Natural Language Processing And How Does It Work", June 30, 2023, https://www.datalinknetworks.net/.
2. Wang, H., Bai, X., Tan, J. et al. Deep prototypical networks-based domain adaptation for fault diagnosis. J Intell Manuf 33, 973–983 (2022). https://doi.org/10.1007/s10845-020-01709-4
3. Wankhade, M., Rao, A.C.S. & Kulkarni, C. A survey on sentiment analysis methods, applications, and challenges. Artif Intell Rev 55, 5731–5780 (2022). https://doi.org/10.1007/s10462-022-10144-1.
4. Sampathirao Suneetha, Prof. S. Viziananda Row, " Aspect-Based Sentiment Analysis: A Comprehensive Survey of Techniques And Applications", Journal of Data Acquisition and Processing, 2023, **38** (3): 177-203.
5. M. Hu and B. Liu, "Mining and summarizing customer reviews," in Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2004, pp. 168–177.
6. Titov and R. McDonald, "A joint model of text and aspect ratings for sentiment summarization," in proceedings of ACL-08: HLT, 2008, pp. 308–316.
7. T. T. Thet, J.-C. Na, and C. S. Khoo, "Aspect-based sentiment analysis of movie reviews on discussion boards," Journal of information science, vol. 36, no. 6, pp. 823–848, 2010.
8. S. Poria, E. Cambria, and A. Gelbukh, "Aspect extraction for opinion mining with a deep convolutional neural network," Knowledge-Based Systems, vol. 108, pp. 42–49, 2016.
9. H. Xu, B. Liu, L. Shu, and S. Y. Philip, "Double embeddings and CNN-based sequence labeling for aspect extraction," in Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), 2018, pp. 592–598.
10. H. Xu, B. Liu, L. Shu, and S. Y. Philip, "Bert post-training for review reading comprehension and aspect-based sentiment analysis," 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), 2019, pp. 2324–2335.
11. Rietzler, S. Stabinger, P. Opitz, and S. Engl, "Adapt or get left behind: Domain adaptation through bert language model fine-tuning for aspecttarget sentiment classification," in Proceedings of the 12th Language Resources and Evaluation Conference, 2020, pp. 4933–4941.
12. Zhang, Wenxuan, Xin Li, Yang Deng, Lidong Bing and Wai Lam. "A Survey on Aspect-Based Sentiment Analysis: Tasks, Methods, and Challenges." ArXiv abs/2203.01054 (2022)
13. Mao, Yue, Yi Shen, Chao Yu and Longjun Cai. "A Joint Training Dual-MRC Framework for Aspect Based Sentiment Analysis." AAAI Conference on Artificial Intelligence (2021).
14. Türkmen, Hazal, Ekinci, Ekin, & Omurca, Sevinç İlhan (2016). A Novel Method for Extracting Feature Opinion Pairs for Turkish. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 9883 LNAI, 161–171. 10.1007/978-3-319-44748-3_16.
15. Hu, Minqing, & Liu, Bing (2004). Mining and Summarizing Customer Reviews. KDD-2004 - Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, no. November, 168–177. 10.1145/1014052.1014073.
16. Pontiki, Maria, Galanis, Dimitris, Pavlopoulos, John, Papageorgiou, Harris, And Popescu, Ana Maria, & Etzioni, Oren (2005). Extracting Product Features and Opinions from Reviews. HLT/EMNLP 2005 - Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference, 339–346. 10.3115/1220575.1220618.
17. Chauhan, Ganpat & Meena, Yogesh & Gopalani, Dinesh & Nahta, Ravi. (2022). A mixed unsupervised method for aspect extraction using BERT. Multimedia Tools and Applications. 81. 1-26. 10.1007/s11042-022-13023-7.
18. Sinno Jialin Pan, James T. Kwok, and Qiang Yang. 2008. Transfer Learning via Dimensionality Reduction. In Proceedings of the 23rd National Conference on Artificial Intelligence -

Volume 2 (Chicago, Illinois) (AAAI'08). AAAI Press, 677–682.
19. S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang. 2011. Domain Adaptation via Transfer Component Analysis. IEEE Transactions on Neural Networks 22, 2 (2011), 199–210. https://doi.org/10.1109/TNN.2010.2091281.
20. 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 (Lille, France) (ICML'15). JMLR.org, 1180–1189.
21. Konstantinos Bousmalis, George Trigeorgis, Nathan Silberman, Dilip Krishnan, and Dumitru Erhan. 2016. Domain Separation Networks. In Advances in Neural Information Processing Systems, D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett (Eds.), Vol. 29. Curran Associates, Inc.
22. Chunning Du, Haifeng Sun, Jingyu Wang, Qi Qi, and Jianxin Liao. 2020. Adversarial and Domain-Aware BERT for Cross-Domain Sentiment Analysis. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, Online, 4019–4028. https://doi.org/10.18653/v1/2020.acl-main.370
23. Chenggong Gong, Jianfei Yu, and Rui Xia. 2020. Unified Feature and Instance Based Domain Adaptation for Aspect-Based Sentiment Analysis. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics, Online, 7035–7045. https://doi.org/10.18653/v1/2020.emnlp-main.572.
24. 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). Association for Computational Linguistics, Minneapolis, Minnesota, 4171–4186. https://doi.org/10.18653/v1/N19-1423.
25. Ying Ding, Jianfei Yu, and Jing Jiang. 2017. Recurrent neural networks with auxiliary labels for cross-domain opinion target extraction. In Association for the Advancement of Artificial Intelligence. 3436––3442.
26. Vasudev Lal, Arden Ma, Estelle Aflalo, Phillip Howard, Ana Simoes, Daniel Korat, Oren Pereg, Gadi Singer, and Moshe Wasserblat. 2021. InterpreT An Interactive Visualization Tool for Interpreting Transformers. In Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations. Association for Computational Linguistics, Online, 135–142. https://www.aclweb.org/anthology/2021.eacl-demos.17
27. Phillip Howard, Arden Ma, Vasudev Lal, Ana Paula Simoes, Daniel Korat, Oren Pereg, Moshe Wasserblat, and Gadi Singer. 2022. Cross-Domain Aspect Extraction using Transformers Augmented with Knowledge Graphs. In Proceedings of the 31st ACM International Conference on Information & Knowledge Management (CIKM '22). Association for Computing Machinery, New York, NY, USA, 780–790. https://doi.org/10.1145/3511808.3557275.
28. Mampilli, B.S., Anand, D. (2022). Cross Domain Aspect Extraction Using Various Embedding Techniques and Language Models. In: Gunjan, V.K., Zurada, J.M. (eds) Proceedings of the 2nd International Conference on Recent Trends in Machine Learning, IoT, Smart Cities, and Applications. Lecture Notes in Networks and Systems, vol 237. Springer, Singapore. https://doi.org/10.1007/978-981-16-6407-6_22.
29. Ricardo Marcondes Marcacini, Rafael Geraldeli Rossi, Ivone Penque Matsuno, Solange Oliveira Rezende, Cross-domain aspect extraction for sentiment analysis: A transductive learning approach, Decision Support Systems, Volume 114, 2018, Pages 70-80, ISSN 0167-9236,
30. Pedro Marcelino, "Transfer learning from pre-trained models", Published in Towards Data Science, Oct 23, 2018.
31. Andreas Pogiatzis, " NLP: Contextualized word embeddings from BERT", Published in Towards Data Science, Mar 20, 2019.

32. Jay Alammar, " The Illustrated BERT, ELMo, and co.", http://jalammar.github.io/illustrated-bert/.
33. Assaad MOAWAD, "Linear layers explained in a simple way", Published in Data Things, Oct 1, 2019.
34. Kiprono Elijah Koech, " Cross-Entropy Loss Function", Published in Towards Data Science, Oct 2, 2020.
35. Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, et al.. 2020. Transformers: State-of-the-Art Natural Language Processing. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pages 38–45, Online. Association for Computational Linguistics.
36. Loshchilov and F. Hutter. 2019. Decoupled Weight Decay Regularization. In ICLR.
37. Wenya Wang and Sinno Jialin Pan. 2018. Recursive Neural Structural Correspondence Network for Cross-domain Aspect and Opinion Co-Extraction. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics. 1—-11.
38. Oren Pereg, Daniel Korat, and Moshe Wasserblat. 2020. Syntactically Aware Cross-Domain Aspect and Opinion Terms Extraction. In Proceedings of the 28th International Conference on Computational Linguistics. International Committee on Computational Linguistics, Barcelona, Spain (Online), 1772–1777. https://doi.org/10.18653/v1/2020. coling-main.158
39. Wenya Wang and Sinno Pan. 2019. Syntactically-Meaningful and Transferable Recursive Neural Networks for Aspect and Opinion Extraction. Computational Linguistics 45 (10 2019), 1–32. https://doi.org/10.1162/COLI_a_00362.
40. Phillip Howard, Arden Ma, Vasudev Lal, Ana Paula Simoes, Daniel Korat, Oren Pereg, Moshe Wasserblat, Gadi Singer, "Cross-Domain Aspect Extraction using Transformers Augmented with Knowledge Graphs", CIKM '22: Proceedings of the 31st ACM International Conference on Information & Knowledge management October 2022Pages 780–790 https://doi.org/10.1145/3511808.3557275.