INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

Meta-domain Adversarial Contrastive Learning for Alleviating Individual Bias in Self-sentiment Predictions

Zhi Li[1], *Ryu Takeda*[1], *Takahiro Hara*[1]

[1]Osaka University, Japan

li.zhi@ist.osaka-u.ac.jp, rtakeda@sanken.osaka-u.ac.jp, hara@ist.osaka-u.ac.jp

Abstract

Self-sentiment provides direct feedback from users and is vital in accurately evaluating and improving the quality of dialogue systems. However, few studies focus on self-sentiment prediction, and the works on third-party sentiment prediction suffer from two problems when predicting self-sentiments: Self-sentiment annotations are labeled by the speakers themselves, leading to solid individual bias in annotations and a sub-optimal prediction; The hardness of collecting sufficient data with self-sentiment annotations limits the size of the data, resulting in the overfitting problem. This work hence proposes a novel meta-learning domain adversarial contrastive neural network (MetaDACNN) that extracts user-shared prior knowledge and learns user-specific classifiers to handle individual bias and to alleviate overfitting. Experimental results on two public datasets show that MetaDACNN improves the prediction performance and alleviates individual bias compared to state-of-the-art models.

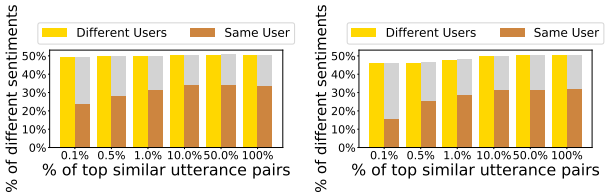
Index Terms: self-sentiment prediction, individual bias, meta-learning, domain adversarial contrastive learning

1. Introduction

Nowadays the interest in developing multimodal dialogue systems (MDS) [1] is on the rise. The recent success of learning methodology across modalities [2, 3, 4] is responsible for that. Compared to traditional dialogue systems which focus on only text-based interactions, MDS has the potential to provide more engaging and effective interactions with users by leveraging the rich and diverse information available in multiple modalities, such as text, speech, images, and videos. In such systems, users' sentiments play vital feedback in evaluating the goodness of responses generated by a dialogue system and enabling the system to better understand and respond to the emotional content of users [5, 6].

Considering the dynamic characteristics of user sentiments in a given dialogue, most previous efforts work on the task of the polarity (positive and negative) prediction for exchange-level sentiments, where the sentiment annotations are labeled per interaction of utterances between the user and the system, i.e., exchange-level dialogue [5, 7, 8, 9]. This work also studies exchange-level sentiment prediction. For the sake of simplicity, all sentiments in the rest of the paper denote the exchange-level sentiment without any special explanation. According to the sentiment annotations labeled by the third-party expert or the speaker themself, the sentiment can be categorized into third-party sentiment and self-sentiment. To our best knowledge, almost all previous studies focus on designing sophisticated models for third-party sentiment prediction [5, 7, 8, 9].

However, these works lose their superiority in self-sentiment prediction (SSP) due to the following two problems: 1) Sentiment expressions vary from person to person, leading to the first problem, i.e., solid individual bias in self-sentiment annotations [3, 10]. The aforementioned works ignore this individual bias and thus result in a sub-optimal prediction. To analyze individual bias, for all the pairs of exchange-level dialogues in the dataset we used, i.e., Hazumi1902 and Hazumi1911[1], we counted the ratio of the pairs that have the most similar features but different self-sentiment annotations[2]. The result is illustrated in Figure 1. As these figures presented, the pairs containing dialogues from different users have over 15% and 18% more cases with different sentiment annotations than those containing dialogues from the same user for Hazumi1902 and Hazumi1911, respectively. The inconsistent relation between features and sentiments makes the model struggle to infer accurate sentiments. 2) Collecting dialogue data with self-sentiment annotations is extremely expensive and time-consuming. This limits the size of data, including the number of users and the number of user-wise data, and results in an over-fitting problem when making predictions with the aforementioned sophisticated models.



(a) *Result on Hazumi1902* (b) *Result on Hazumi1911*

Figure 1: *Ratio of exchange-level dialogues with the most similar features but different sentiment annotations*

To tackle the above two problems, we propose a novel meta-learning domain adversarial contrastive neural network (MetaDACNN) that combines a meta-learning framework and a novel class-wise domain adversarial contrastive learning framework. The base model of MetaDACNN consists of an encoder, a classifier, and a domain discriminator. To handle individual bias, we borrow the idea from gradient-based meta-learning [11] and consider self-sentiment predictions for different users separately. We denote the prediction for one user as one learning task. The meta-learning framework in MetaDACNN locally updates the classifier to learn a meta-classifier that contains user-shared prior knowledge extracted from past SSP tasks.

[1]The doi is doi/10.32130/rdata.4.1

[2]We calculate the cosine similarities with the output of the encoder, i.e., $\mathcal{G}_\phi(x_i)$ in Equation 2.

10.21437/Interspeech.2023-510

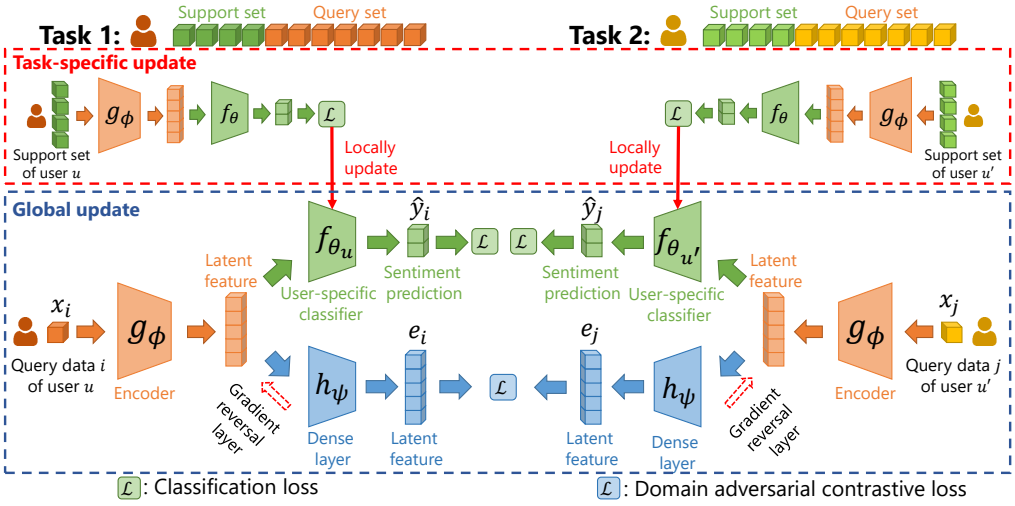


Figure 2: *Overview of the proposed MetaDACNN*

This knowledge can guide the meta-classifier to quickly adapt to similar unseen prediction tasks. Different from traditional meta-learning [11], which locally updates the whole model, MetaDACNN locally updates the classifier only and freezes the encoder to extract user-shared features. These features can guide the meta-classifier to learn user-shared prior knowledge from limited data and to alleviate the over-fitting problem.

Traditional meta-learning requires thousands of learning tasks to learn task-shared knowledge [11]. To adapt to the limited tasks (tens of users) in SSP, we denote the prediction for one user as a domain and introduce domain adversarial learning (DAL) into SSP. DAL is effective to extract domain-shared features from limited domains [12]. Analogously, it is expected to alleviate overfitting by extracting user-shared features from limited users. Traditional DAL methods rely on the user labels to confuse the domain discriminator and to extract domain-shared features [12, 13, 14]. However, all data in one learning task are labeled with the same user ID in meta-learning. We hence develop a new class-wise domain adversarial contrastive learning framework that removes the use of the highly imbalanced user labels by contrastively aligning the latent feature spaces learned from different users' data.

Our experimental results on two public datasets demonstrate the effectiveness of our MetaDACNN in improving the prediction performance and alleviating individual bias compared to state-of-the-art domain adversarial learning and meta-learning methods. Besides, our experimental results also indicate that simply increasing the trainable parameters and using complex models cannot improve classification performance due to the limited training data and solid individual bias.

2. Method

2.1. Problem Formulation

Inspired by meta-learning [11], we consider the self-sentiment prediction for different users as different learning tasks. To study the self-sentiment prediction, we leverage two public datasets, Hazumi1902 and Hazumi1911, which are composed of exchange-level dialogues from a dialogue system. Each exchange-level dialogue includes the multimodal input and the corresponding self-sentiment annotation. We denote a dataset as $\mathcal{D} = \{\mathcal{D}_u | u \in U\}$, where U is the user set and $\mathcal{D}_u = \{(x_i, y_i) | i \in N_u\}$ is the set of exchange-level dialogues for user u. Here, x_i, y_i, and N_u are the multimodal input (a high dimensional vector), the binary self-sentiment annotation labeled by user u, and the number of exchange-level dialogues for user u, respectively. To handle the individual bias in annotations, we learn a user-specific classifier $\mathcal{F}_{\theta_u}(\cdot)$ for each user u to adapt to the individual bias, where θ_u is the parameter set of the classifier. For u, The predictive self-sentiment annotation $\hat{y}_i$ is calculated by

$$\hat{y}_i = \mathcal{F}_{\theta_u}(x_i). \tag{1}$$

2.2. Overview of Proposed Method

To tackle the individual bias problem and the overfitting problem caused by the limited data, we propose a new meta-learning domain adversarial contrastive neural network (MetaDACNN), which is depicted in Figure 2. MetaDACNN combines a meta-learning framework and a novel domain adversarial learning framework. Motivated by [11], the meta-learning framework includes a user-specific update and a global update. The former locally updates a user-shared classifier to generate user-specific classifiers that can handle individual bias. The latter learns the user-shared classifier and combines a class-wise domain adversarial contrastive learning framework that explicitly aligns the feature spaces learned from different users' data to extract user-invariant features for alleviating overfitting.

2.3. Base Model

To learn user-specific classifiers and a user-shared encoder, we add an encoder $\mathcal{G}_\phi(\cdot)$ and reformulate the prediction of self-sentiment as

$$\hat{y}_i = \mathcal{F}_\theta(\mathcal{G}_\phi(x_i)), \tag{2}$$

where $\mathcal{F}_\theta$ is the user-shared classifier with the parameter θ. The input x_i has different pre-processed unimodal features, including audio a_i: 384 dim, linguistic l_i: 768 dim, and video v_i: 86 dim. Considering the small size of the dataset, we follow the same early fusion method in [3, 10] to calculate x_i by concatenating different unimodal features $x_i = [a_i, l_i, v_i]$.

2.4. User-specific update

To handle the individual bias in self-sentiment annotations, we learn a user-specific classifier for each user in this part. To achieve this, we split each user's dataset $\mathcal{D}_u$ into a support set $\mathcal{S}_u$ and a query set $\mathcal{Q}_u$ in chronological order. Then, the support set $\mathcal{S}_u$ is leveraged to locally update the user-shared classifier $\mathcal{F}_\theta$, which is formulated by

$$\theta_u \leftarrow \theta - \alpha \cdot \nabla_\theta \mathcal{L}\left(\mathcal{S}_u | \mathcal{F}_\theta, \mathcal{G}_\phi\right), \tag{3}$$

where α is the learning rate for this local update. The loss $\mathcal{L}\left(\mathcal{S}_u | \mathcal{F}_\theta, \mathcal{G}_\phi\right)$ denotes the binary cross-entropy (BCE) loss measured on $\mathcal{S}_u$ and is calculated by

$$\mathcal{L}\left(\mathcal{S}_u | \mathcal{F}_\theta, \mathcal{G}_\phi\right) = -\frac{1}{|S_u|} \sum_{i=1}^{|\mathcal{S}_u|} \mathcal{L}_{BCE}(\hat{y}_i, y_i). \tag{4}$$

By doing so, we get the user-specific classifier $\mathcal{F}_{\theta_u}$.

2.5. Domain Contrastive Global Update

In this part, we learn the user-shared encoder and the user-shared classifier that contain prior knowledge and can fast adapt to unseen users. To better extract prior knowledge from limited training users, we develop a new class-wise domain adversarial contrastive learning framework to explicitly align different users' feature spaces learned by the encoder $\mathcal{G}_\phi$.

Following [15, 16], we add an additional tiny network $\mathcal{H}_\psi$ with several dense layers to get the latent feature for domain adversarial learning. The latent feature of data x_i is formulated as $e_i = \mathcal{H}_\psi(\mathcal{G}_\phi(x_i))$. After that, we propose a novel class-wise contrastive loss that aligns different users' latent feature spaces by punishing the difference of features between the instances from different users. For every two users (u, u') in the dataset, we calculate the contrastive loss on their query sets. The contrastive loss $\mathcal{L}_{DAC}(\mathcal{Q}_u, \mathcal{Q}_{u'} | \mathcal{G}_\phi, \mathcal{H}_\psi)$ of (u, u') is formulated by

$$\mathcal{L}_{DAC} = -\frac{1}{|\mathcal{Q}_u|} \sum_{i=1}^{|\mathcal{Q}_u|} \frac{1}{|\mathcal{Q}_{u'}|} \sum_{j=1}^{|\mathcal{Q}_{u'}|} \mathbb{1}_{[y_i = y_j]} \log \frac{\exp\left(\mathrm{sim}\left(e_i, e_j\right)/\tau\right)}{\sum_{k=1}^{|\mathcal{Q}_u|} \mathbb{1}_{[i \neq k]} \mathbb{1}_{[y_i = y_k]} \exp\left(\mathrm{sim}\left(e_i, e_k\right)/\tau\right)}, \tag{5}$$

where $\mathbb{1}$ is the indicator. $\mathrm{sim}(\cdot)$ denotes the cosine similarity function and τ controls the temperature. Borrowing the idea from [12], we achieve the punishment of the domain difference by employing a gradient reversal layer $\mathrm{GRL}(\cdot)$ when calculating the latent feature e_i, which is reformulated as

$$e_i = \mathcal{H}_\psi(\mathrm{GRL}(\mathcal{G}_\phi(x_i))). \tag{6}$$

Besides, we also measure the self-sentiment prediction loss on the query sets of u and u' with user-specific classifiers $\mathcal{F}_{\theta_u}$ and $\mathcal{F}_{\theta_{u'}}$, respectively. Finally, the total loss $\mathcal{L}(\mathcal{Q}_u, \mathcal{Q}_{u'} | \mathcal{F}_\theta, \mathcal{G}_\phi, \mathcal{H}_\psi)$ in this part is given by

$$\begin{aligned} \mathcal{L} =& \mathcal{L}(\mathcal{Q}_u | \mathcal{F}_{\theta_u}) + \mathcal{L}(\mathcal{Q}_{u'} | \mathcal{F}_{\theta_{u'}}) \\ &+ \lambda \cdot \mathcal{L}_{DAC}(\mathcal{Q}_u, \mathcal{Q}_{u'} | \mathcal{G}_\phi, \mathcal{H}_\psi), \end{aligned} \tag{7}$$

where λ is a hyper-parameter that balances the impact of the domain adversarial loss. After that, the user-shared classifier $\mathcal{F}_\theta$, the encoder $\mathcal{G}_\phi$, and the tiny network $\mathcal{H}_\psi$ are trained by

$$\begin{aligned} \{\theta, \psi\} &\leftarrow \{\theta, \psi\} - \beta \cdot \nabla_{\{\theta,\psi\}} \mathcal{L}(\mathcal{Q}_u, \mathcal{Q}_{u'} | \mathcal{F}_\theta, \mathcal{G}_\phi, \mathcal{H}_\psi), \\ \phi &\leftarrow \phi + \beta \cdot \nabla_\phi \mathcal{L}(\mathcal{Q}_u, \mathcal{Q}_{u'} | \mathcal{F}_\theta, \mathcal{G}_\phi, \mathcal{H}_\psi), \end{aligned} \tag{8}$$

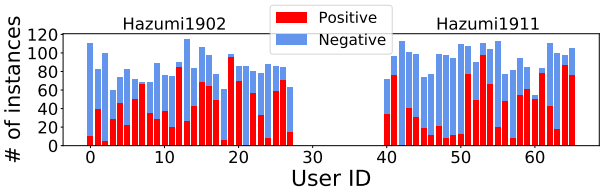


Figure 3: *User-wise label distribution*

where β is the learning rate for this global update. Here, the encoder $\mathcal{G}_\phi$ is optimized by the gradient ascent after adding the gradient reversal layer $\mathrm{GRL}(\cdot)$ [12]. By doing so, domain differences are punished to help $\mathcal{G}_\phi$ learn user-shared features.

2.6. Self-sentiment Prediction

When making self-sentiment predictions for a user u, the learned MetaDACNN conducts user-specific update on this user's support set $\mathcal{S}_u$ to get the user-specific classifier $\mathcal{F}_{\theta_u}$ by Equation (3) and (4). Then, the self-sentiment annotation of any instance x from u is predicted by $\hat{y} = \mathcal{F}_{\theta_u}(\mathcal{G}_\phi(x))$.

3. Experiment

3.1. Experiment Setting

Dataset. We used Hazumi1902 and Hazumi1911 datasets. These datasets consist of 2,337 and 2,439 exchange-level dialogue instances from 28 and 26 users, respectively. Each instance contains the multimodal feature and a binary self-sentiment annotation, where the multimodal feature includes audio, linguistic, and video features. The number of positive and negative instances in these datasets is highly imbalanced at the user level. We counted the user-wise number of the positive and the negative instances for users in Hazumi1902 and Hazumi1911 and put the result in Figure 3 to illustrate this phenomenon. To alleviate the negative impact of the imbalanced annotations, we apply the SMOTE method [17] to over-sample the minority instances for each training user. Moreover, we performed a 5-fold cross-validation to split users into training and test users for Hazumi1902 and Hazumi1911.

Evaluation Metrics. To accurately verify the performance on self-sentiment prediction with the highly imbalanced labels, we employed the widely used area under the receiver operating characteristic curve (AUC) metric. For each test user, we locally updated the trained user-shared classifier with the user's support set and measured the AUC score for instances in the user's query set with the updated user-specific classifier.

Comparison Methods. We compared our MetaDACNN with two baselines: the support vector classifier (SVC) [18] and the vanilla neural network (VNN), where VNN is our base model without the meta-learning and the domain adversarial learning frameworks. We further compared MetaDACNN with the representative meta-learning method (i.e., MAML [11]) and the following state-of-the-art multi-domain adversarial learning methods: DANN [12], MAN-L2 [13], MAN-NLL [13], and MADA [19]. Besides, we added a variance of MetaDACNN, named MetaDANN, that replaces our class-wise domain adversarial contrastive learning framework with the domain adversarial learning in DANN. For fair comparisons, we aligned the backbone of the base model for all neural network methods.

Implementation Details. All NN methods were implemented by Pytorch [20]. SVC was the model with the default hyper-

Table 1: *Comparison results on AUC are reported. Bold shows the winner.*

	Hazumi1902								
	SVC	VNN	DANN	MAN-L2	MAN-NLL	MADA	MAML	MetaDANN	MetaDACNN (ours)
Fold-1	0.617	0.620	0.602	0.627	0.624	0.584	0.637	0.654	**0.655**
Fold-2	0.644	0.633	0.610	0.629	0.632	0.614	0.664	0.660	**0.672**
Fold-3	**0.639**	0.626	0.581	0.586	0.586	0.573	0.623	0.620	0.627
Fold-4	0.597	0.630	0.644	**0.687**	0.681	0.677	0.665	0.680	0.665
Fold-5	0.610	0.603	0.605	0.600	0.599	0.584	0.590	**0.626**	0.621
Average	0.622	0.622	0.608	0.626	0.624	0.606	0.636	**0.648**	**0.648**
	Hazumi1911								
	SVC	VNN	DANN	MAN-L2	MAN-NLL	MADA	MAML	MetaDANN	MetaDACNN (ours)
Fold-1	**0.696**	0.646	0.615	0.678	0.676	0.681	0.677	0.643	0.649
Fold-2	0.697	0.689	0.638	0.676	0.677	0.689	0.685	0.716	**0.723**
Fold-3	0.751	0.684	0.771	0.773	0.772	0.735	0.652	0.618	**0.779**
Fold-4	0.524	0.488	0.458	0.468	0.471	0.463	0.523	0.522	**0.536**
Fold-5	0.639	0.649	0.672	0.681	0.68	0.636	0.653	0.697	**0.701**
Average	0.661	0.631	0.631	0.655	0.655	0.641	0.638	0.639	**0.677**

Table 2: *Result of the comparison between unimodal and multimodal features. The average AUC of 5-fold cross-validation is reported. Bold shows the winner.*

Dataset	Feature	SVC	VNN	MetaDACNN
Hazimi 1902	Linguistic	**0.631**	0.605	0.646
	Audio	0.605	0.576	0.609
	Video	0.551	0.526	0.521
	Multimodal	0.622	**0.622**	**0.648**
Hazimi 1911	Linguistic	0.644	0.602	0.638
	Audio	0.654	**0.654**	**0.694**
	Video	0.599	0.583	0.541
	Multimodal	**0.661**	0.631	0.677

parameters and the probability output implemented by Scikit-learn [21]. The base model for all NN methods consists of an encoder and a classifier, where the encoder has 2 dense layers with the shape of $[256, 64]$ and the classifier has 3 dense layers with the shape of $[32, 16, 2]$. The tiny network of MetaDACNN has 2 dense layers with the shape of $[32, 16]$. We experimentally set the local learning rate α, the global learning rate β, and the balancing factor λ to 0.00001, 0.0001, and 0.1 respectively. We applied the mini-batch trick to accelerate our training. The mini-batch size was 128 for all non-meta-learning methods. We used the mini-batch trick at the user level for all meta-learning methods and set the mini-batch size to 8 users. For each training user, we over-sampled this user's data and employed 5 positive instances and 5 negative instances as the support set, where the rest data serves as the query set.

3.2. Experiment Results

Comparison of different methods. The comparison results on Hazumi1902 and Hazumi1911 are listed in Table 1. From this table, we find that: (1) MetaDACNN outperforms other comparisons w.r.t. AUC in most cases and gains an average 1.89% (2.42%) improvement in Hazumi1902 (Hazumi1911) compared to the bast baseline. This is because MetaDACNN effectively handles the individual bias issue by extracting user-shared prior knowledge and learning user-specific classifiers from limited training data. (2) SVC outperforms most NN-based methods, including domain adversarial methods (DANN, MAN-L2, MAN-NLL, MADA) and the meta-learning method (MAML). Since NN-based methods have more trainable parameters than SVC, this result indicates that simply increasing the trainable parameters and using complex models cannot improve classification performance, especially with a small size of dataset and solid individual bias in annotations. (3) MetaDACNN outperforms MetaDANN on Hazumi1911 and achieves a comparable performance to MetaDANN on Hazumi1902. This is because Hazumi1911 contains more annotations with individual bias (the yellow bar in Figure 1) compared to Hazumi1902, and our class-wise domain adversarial contrastive learning framework is more effective in handling individual bias. Besides, Hazumi1902 contains more annotations with noises (the brown bar in Figure 1) compared to Hazumi1911. The comparison results in Hazumi1902 indicate that MetaDACNN will lose its superiority when the annotation noise increases.

Comparison between unimodal and multimodal features. To study the effectiveness of different methods in fusing multimodal features, we employed SVC, VNN, and MetaDACNN and compared their self-sentiment prediction performances with unimodal and multimodal input. Table 2 reports the result. We see that the early fusing method used in our MetaDACNN gains a few or even negative improvements from the best unimodal input to multimodal input. This result demonstrates the hardness of fusing different types of features, especially with a small dataset and strong individual biases, leaving a future work to design better methods for multimodal feature fusion.

4. Conclusion

This work proposed a new method, named MetaDACNN, for self-sentiment prediction. To alleviate the individual bias and the overfitting problems, MetaDACNN combines a meta-learning framework and a novel class-wise domain adversarial contrastive framework to effectively learn user-shared prior knowledge that can be fast updated to user-specific classifiers with only limited data. Our experimental results on two real-world datasets demonstrate the superiority of MetaDACNN.

5. References

[1] M. Firdaus, H. Chauhan, A. Ekbal, and P. Bhattacharyya, "Emosen: Generating sentiment and emotion controlled responses in a multimodal dialogue system," *IEEE Transactions on Affective Computing*, vol. 13, no. 3, pp. 1555–1566, 2022.

[2] S. Yang, R. Zhang, S. M. Erfani, and J. H. Lau, "Unimf: A unified framework to incorporate multimodal knowledge bases intoend-to-end task-oriented dialogue systems," in *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, Z. Zhou, Ed., 2021, pp. 3978–3984.

[3] W. Wei, S. Li, S. Okada, and K. Komatani, "Multimodal user satisfaction recognition for non-task oriented dialogue systems," in *ICMI '21: International Conference on Multimodal Interaction*, 2021, pp. 586–594.

[4] Y. Hirano, S. Okada, and K. Komatani, "Recognizing social signals with weakly supervised multitask learning for multimodal dialogue systems," in *ICMI '21: International Conference on Multimodal Interaction*, 2021, pp. 141–149.

[5] H. Takatsu, R. Ando, Y. Matsuyama, and T. Kobayashi, "Sentiment analysis for emotional speech synthesis in a news dialogue system," in *Proceedings of the 28th International Conference on Computational Linguistics*, 2020, pp. 5013–5025.

[6] M. Wankhade, A. C. S. Rao, and C. Kulkarni, "A survey on sentiment analysis methods, applications, and challenges," *Artificial Intelligence Review*, vol. 55, no. 7, pp. 5731–5780, 2022.

[7] C. Lin, S. Zhao, L. Meng, and T. Chua, "Multi-source domain adaptation for visual sentiment classification," in *The Thirty-Fourth AAAI Conference on Artificial Intelligence*, 2020, pp. 2661–2668.

[8] Y. Dai, J. Liu, X. Ren, and Z. Xu, "Adversarial training based multi-source unsupervised domain adaptation for sentiment analysis," in *The Thirty-Fourth AAAI Conference on Artificial Intelligence*, 2020, pp. 7618–7625.

[9] D. Bertero, F. B. Siddique, C. Wu, Y. Wan, R. H. Y. Chan, and P. Fung, "Real-time speech emotion and sentiment recognition for interactive dialogue systems," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2016, pp. 1042–1047.

[10] W. Wei, S. Li, and S. Okada, "Investigating the relationship between dialogue and exchange-level impression," in *International Conference on Multimodal Interaction, ICMI 2022*, 2022, pp. 359–367.

[11] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proceedings of the 34th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 70. PMLR, 2017, pp. 1126–1135.

[12] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. S. Lempitsky, "Domain-adversarial training of neural networks," *Journal of Machine Learning Research*, vol. 17, pp. 59:1–59:35, 2016.

[13] X. Chen and C. Cardie, "Multinomial adversarial networks for multi-domain text classification," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2018, pp. 1226–1240.

[14] A. S. Sebag, L. Heinrich, M. Schoenauer, M. Sebag, L. F. Wu, and S. J. Altschuler, "Multi-domain adversarial learning," in *7th International Conference on Learning Representations*, 2019.

[15] M. Ye, X. Zhang, P. C. Yuen, and S. Chang, "Unsupervised embedding learning via invariant and spreading instance feature," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6210–6219.

[16] T. Chen, S. Kornblith, M. Norouzi, and G. E. Hinton, "A simple framework for contrastive learning of visual representations," in *Proceedings of the 37th International Conference on Machine Learning*, vol. 119, 2020, pp. 1597–1607.

[17] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.

[18] J. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," *Advances in Large-Margin Classifiers*, vol. 10, 06 2000.

[19] Z. Pei, Z. Cao, M. Long, and J. Wang, "Multi-adversarial domain adaptation," in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, 2018, pp. 3934–3941.

[20] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Z. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems*, 2019, pp. 8024–8035.

[21] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. VanderPlas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.