2023 IEEE International Conference on Software Analysis, Evolution and Reengineering (SANER) | 978-1-6654-5278-6/23/$31.00 ©2023 IEEE | DOI: 10.1109/SANER56733.2023.00117

Multimodal Sentiment Analysis under modality deficiency with prototype-Augmentation in software engineering

Baolei Wang
School of Software, Yunnan University
Kunming, China
baoleicheerup@163.com

Xuan Zhang
School of Software, Yunnan University
Yunnan Key Laboratory of Software Engineering
Kunming, China
zhxuan@ynu.edu.cn

Kunpeng Du
School of Software, Yunnan University
Kunming, China
dkp0801@mail.ynu.edu.cn

Chen Gao
School of Information Science and Engineering, Yunnan University
Kunming, China
929165733@qq.com

Linyu Li
School of Software, Yunnan University
Kunming, China
970800412@qq.com

***Abstract*—Sentiment analysis has a wide range of promising applications in software engineering, and the development of deep learning has demonstrated that the uniform representation of different modalities can improve the model performance of sentiment analysis. However, in practical applications, multimodal sentiment analysis always faces unsatisfactory situations, especially when the modality has missing samples, most models may fail. For example, social dynamics of technicians in developer communities can face modality unavailability due to privacy settings. Several existing works based on deep learning and regularization methods have explored the modal missing problem, but these works cannot balance the cases of modal general missing (rate < 50%) and severe missing (rate ≥ 50%), and do not consider the resource consumption during model inference. Therefore, in this paper, we proposed a prototype augmented multimodal teacher-student network (PAMD) to address the above issues. Specifically, a multi-level and multi-origin distillation strategy is used to minimize the required resources and inference time, and prototype augmentation is used to guarantee the performance of the model when a modality is severely missing. Extensive experiments are conducted on different benchmark datasets to explore a network that balances performance and resource consumption. And It achieves good results in different modalities of missing cases.**

Keywords—Multimodal Sentiment Analysis, software engineering, Prototype augmented, Distillation, Severe Missing

I. INTRODUCTION

Multimodal Sentiment Analysis (MSA) is widely used in all areas of software engineering and is used throughout the software engineering lifecycle. For example, as one of the most important parts of the software engineering lifecycle, software development is a highly collaborative activity that is highly influenced by the emotional state of developers. Negative affective states can make developers less productive in software projects and can easily introduce software defects [33], while positive affect is expected to increase developer productivity [1]. In addition, obtaining valid sentiment status from APP reviews, technical Q&A sites such as Stack Overflow, and developers' comments on APIs is crucial for subsequent product improvement and service refinement. For example, the comment "**I am not able to deploy my App Engine project locally.**" on the Java API predicts a negative sentiment, which in turn leads to the possibility of optimizing the mentioned "**App Engine**" service. This shows that it is necessary and essential to conduct research on sentiment analysis in software engineering.

However, for some sentiment analysis tasks in software engineering, there are no annotated datasets available, and manual annotation of datasets is time-consuming and labor-intensive. In this case, an alternative approach is to train similar tasks with uniform goals from existing datasets [2]. With the explosive growth of information, the use of unimodal information for sentiment analysis is becoming increasingly inadequate, and therefore multimodal sentiment analysis is coming to the forefront. Related studies have shown that combining different modalities can learn complementary features, resulting in better joint multimodal representations [3], and sentiment analysis tasks benefit from this. However, previous works [4,5] are mostly based on an assumption that the modality is complete and available, while in reality, due to device or privacy constraints, more often face the case of missing modality. Fig.1 shows the possible cases of missing modality encountered when introducing Zookeeper, a distributed application coordination service software. These include missing transcribed text due to unclear sound, missing audio due to noise, and missing video modality due to light. Therefore, this paper focuses on how to achieve comparable performance to the full modal model in the absence of modality.

Early work [6, 7] mainly tackled the missing problem by directly losing the missing modes or using matrices to estimate the missing modes, an approach that degrades the overall performance to some extent. In recent years, with the rapid development of deep learning, researchers have started to use neural networks to learn the potential relationships between modalities. MCTN [21] uses cyclic transformations between modalities to generate information about other modalities from only one modality. Based on the above work, [8,9,10] developed learning models based on Transformer for different missing cases. However, none of the above works consider the key challenge of models in the inference process: high computational resources. Ma et al. [11] focuses on the model's

2640-7574/23/$31.00 ©2023 IEEE
DOI 10.1109/SANER56733.2023.00117

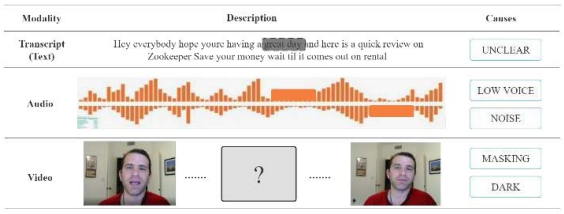

Fig. 1. Example of missing modal case, where shading represents missing modal information

flexibility and efficiency in severe modal missing cases while ignoring the performance of modal general missing models. Moreover, the model is based on the construction of Bayesian networks for meta-learning, which is more complex to handle and has poor generalization characteristics. Therefore, this paper is a cutting-edge research work on modal missing in the field of software engineering that mainly addresses the two challenges mentioned above, i.e., balancing general missing (rate <50%) and severe missing of modalities with controlled resource consumption.

In general, this paper addresses the problem of "less" in multimodal sentiment analysis: how to reduce resource consumption and inference time and how to achieve the same prediction results with less data (generally missing and severely missing), especially when the missing cases are more severe. The contributions are as follows:

1. We innovatively propose a multi-level and multi-origin distillation strategy to distill different knowledge into the student network by differentially constructing a teacher network to explore ways to minimize the required resources and reasoning time. The network is constructed based on Transformer, which is able to reconstruct the impact of modalities deficiency in different situations.
2. A sample prototype is constructed with the help of meta-learning design ideas for enhancing the missing modal representation under severe modal deficits. Overall, the network is able to reduce resource consumption and inference time while ensuring model performance.
3. The proposed model PAMD shows good performance on both benchmark multimodal sentiment classification and software engineering datasets. The experimental results show that the proposed model is able to balance performance and resource consumption in the face of different cases of modality deficiency.

II. RELATED WORK

A. Multimodal Sentiment Analysis under modality deficiency

Multimodal data establishes the data foundation for intelligent software development. Multimodal sentiment analysis aims to predict people's emotions from multimodal data such as video, audio and text. Earlier work [12-14] focused on aligned multimodal data, which implies that there must be an explicit correspondence in each modality. However, due to the heterogeneity of the data, the main challenge is how to exploit their complementarity to obtain a unified representation of the data in different modalities. For this purpose, researchers first proposed early fusion strategies [15], while recently starting to explore late fusion [16]. However, the aforementioned models are not very usable for more realistic modal deficiency scenarios.

Related research on dealing with the problem of missing sentiment modalities can usually be divided into three categories: data augmentation methods [17], generative methods, and joint learning methods. Traditional generative methods include AE [19], DAL [20], and cycleGAN [30]. Tran [18] proposed a cascaded residual autoencoder (CRA) to exploit the residual mechanism on the autoencoder structure, which can acquire corrupted data and estimate the function to recover incomplete data well. And joint learning methods try to learn the joint representation based on the relationship between different modalities. Yuan [8] extracted intra- and inter-modal relations using Transformer and designed a Transformer-based feature reconstruction network to reproduce the semantics of missing modalities. Zhao [9] also applied cycle consistency learning to the reconstruction, in which a cascaded residual autoencoder was CRA-based cross-modal. Zeng [10] added a tag encoding module to label the missing modes based on the above work to cope with the general problem of missing modes. However, none of the above works include the non-negligible computational resources in MSA to measure the model's performance.

B. The Computational cost in Multimodal Machine Learning

In recent years, the sequence-to-sequence model based on Transformer [31] has been widely studied in the multimodal domain. Tsai [26] proposed a multimodal transformer (MulT) that includes a cross-modal attention mechanism to learn representations from unaligned multimodal data. The model is able to infer missing modal semantics based on existing modal dependencies without explicitly aligning data. However, almost all transformer-based multimodal models face high computational resource problems due to modal heterogeneity and complex inference processes [23]. Multimodal end-to-end models with sparse cross-modal attention mechanisms have recently been proposed to reduce computational overhead [25]. Multimodal end-to-end models with sparse cross-modal attention mechanisms have recently been proposed to reduce computational overhead [25]. Knowledge distillation [22], which transfers knowledge from one deep learning model (teacher) to another (student), was initially used to reduce the distance between the probability distributions of the output classes of two networks [24], while today's combination with neural networks shows promising results. Therefore, we propose to use an approach that employs a Multi-level and Multi-origin distillation strategy and prototype enhancement to minimize the required resources and inference time.

C. Sentiment analysis for Software Engineering

Sentiment analysis has a wide range of applications in the field of software engineering. On the one hand, sentiment analysis techniques can be used to analyze developers' perceptions of software products (e.g., APIs, etc.) to assist in the construction of recommendation systems in software engineering. For example, Lin [34] analyzed the sentimental attitude of developers towards different APIs on StackOverflow and used it as the basis for the recommendation system for APIs. On the other hand, sentiment analysis techniques can be used to perceive developers' sentiments so that researchers can explore the interaction between developers' sentiments and the software development process. For example, we can investigate the correlation between the time factor and developer sentiment,

and Ortu [35] investigated the correlation between the length of problem-solving time and developer sentiment. Such work uses sentiment analysis techniques to help researchers better understand developers' emotions and their behavioral patterns, thereby targeting developers to improve their development efficiency. However, all the above-mentioned works use sentiment text information to accomplish sentiment analysis tasks, and there is less exploration of multimodal sentiment analysis.

III. METHODOLOGY

A. The Proposed Framework PAMD

In this section, we focus on describing the approach to learning robust representations for different modal deficits through multimodal distillation and prototype augmentation. PAMD (Prototype Augmentation Multimodal Distillation) is divided into three main submodules: the Teacher-Student Network module (Section 3.2), the multi-level multi-origin Distillation module (Section 3.3), and the Prototype Assist module (Section 3.4). The general framework is shown in Fig.2.

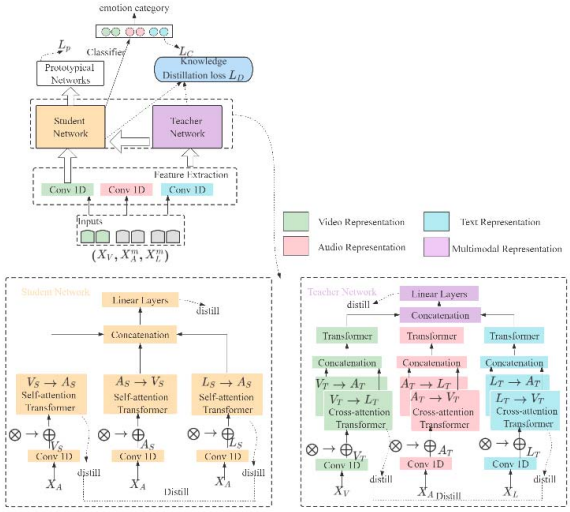

Fig. 2. Overall framework. Different modalities are extracted by 1D convolutional layer features and input to the teacher-student network to complete the subsequent process. Where X_A^m and X_V^m denote audio modality and language modality missing, respectively. One type of multimodal distillation is shown in the following figure,distillation of knowledge from the teacher network through video and text modalities to the student network.

Our overall goal is to determine the speaker's emotional state through incomplete modalities and reduce resource consumption while maintaining performance. Each video clip contains three modalities: visual (V), audio (A), and language (L). We represent $\{X_V, X_A, X_L\}$ as multimodal sequence outputs through the embedding layer, each with a different feature dimension and time series, which can be expressed as $X_i = \zeta^{s_i \times d_i}$, where s_i denotes the sequence length and d_i denotes the feature vector dimension. To ensure the requirements of the subsequent attention mechanism dot product computation, we make the different modalities have the same dimensionality by means of a 1D convolution layer:

$$\hat{X}_i = Conv1dX_i = \zeta^{s_i \times d_i} \tag{1}$$

B. Teacher-student network based on Transformer

Inspired by Tsai [26], the teacher and student networks take as input a multimodal sequence $\{X_V, X_A, X_L\}$, the teacher network uses a cross-modal transformer so that one modality can receive information from another; and the student network uses a transformer based on a self-attentive mechanism. In order to make the input to the cross-modal transformer information time-ordered, the positional embedding (PE) is added to the output $\hat{X}_i$ of the convolutional layer:

$$P_{(V,A,L)}^{[0]} = X_{(V,A,L)} + PE\left(F_{(V,A,L)}, d\right) \tag{2}$$

Where $PE\left(F_{(V,A,L)}, d\right)$ can compute the index of each position for embedding, and $P_{(V,A,L)}^{[0]}$ denotes the low-dimensional position features.

Since the data to be processed contains three modalities: video, audio, and language (text), we propose to use multiple transformers to apply cross-attention to each combination of query (Q), key (K), and value (V) pairs, as well as a transformer based on the self-attention mechanism in the student network. Fig. 2 The following figure shows one architecture of the teacher network, where $V_T \to L_T$ denotes the teacher network with modal transfer, Q from modality V (video), and K and V from modality L (language). Based on this definition, $A_T \to V_T$ and $A_T \to L_T$ constitute the audio branch of the teacher network since they both use the audio modality as Q. Similarly, we also define the video branch and the language branch of the student network. The complete teacher network contains the above three branches. For the student network, we use only the audio modality as input and model the missing language and video modalities with $V_S \to A_S$ and $L_S \to A_S$, respectively. In addition, we provide a cross-modal potential adaptation to fuse cross-modal information. In the layer i cross-modal attention block, the input $X_V \in \mathbb{R}^{S_V \times d_V}$ to the cross-modal attention computation is the output of the layer i-1 cross-modal attention block then obtained by layer normalization, and the input $X_A \in \mathbb{R}^{S_A \times d_A}$ is the cross-modal attention computation obtained by layer normalization of the origin modalities as follows:

$$\xi_V = softmax\left(\frac{X_V W_{Q_V} W_{K_A}^T X_{K_A}^T}{\sqrt{d_k}}\right) X_A W_{V_A} \tag{3}$$

Where the query $Q_V = X_V W_{Q_V}$ comes from modality V (video), the key (key), and the sum value (value) come from modality A (audio). $\sqrt{d_k}$ represents the scale of the softmax computed fraction matrix scaling. Each modality continuously updates its sequence by interacting with the underlying information of other modalities through the multi-head cross-modal attention module.

C. Multi-level and multi-origin distillation

For multi-origin distillation, we conducted experiments on three teacher configurations. One is the complete teacher network with a random combination of two branches; the other has unimodal branches: video, audio, and language branches. The last sequence element in the output sequence of the different transformers is passed through a linear layer, respectively. The student network is a twin network identical to the full teacher network; the other setups are simplified

versions of this network with detailed configurations as described in Section IV.

In addition, since different layers in the deep network carry different information, we fix the distillation sources and apply distillation to each stage of the network, as shown in Fig. 3. Cross entropy is applied to calculate distillation loss in a high-level feature final linear layer and two low-level feature post-attention layers, the Attention Map layers. In contrast to the teacher network, the student network can only learn from incomplete modalities. Extracting different Attention Map from multiple transformer can explain the missing modality in the student network to some extent. For simplicity of representation, similar to Equation 3, we define $\beta \to \alpha$ as the sequence β as Q, and K, V from the sequence α.

$$\chi(\beta \to \alpha, t) = softmax(\frac{(Q_\beta K_\alpha^T)}{\sqrt{d_k}}, t) \tag{4}$$

$$o^{\beta_S \to \alpha_S} = [\chi(\beta_S \to \alpha_S)]_{i=1}^{i=l} \tag{5}$$

$$o^{\beta_T \to \alpha_T} = [\chi(\beta_T \to \alpha_T)]_{i=1}^{i=l} \tag{6}$$

We use Eq. (4–6) to obtain the attention map from all transformer layers, where t is the parameter regulating the temperature distribution and o denotes the storage variable of the multilayerχ. The distillation loss of a pair is calculated using Eq. (7) and (8), and the final distillation loss is calculated by averaging the $\mathcal{L}_D$obtained from all pairs by Eq. (9).

$$\mathcal{L}_D = \frac{\sum_{i=1}^{l} \sum_{j=1}^{m} L(o_{ij}^{\beta_S \to \alpha_S}, o_{ij}^{\beta_T \to \alpha_T})}{ml} \tag{7}$$

$$L(a, b) = -alog(b) \tag{8}$$

$$\mathcal{L}' = \frac{\sum_{i=1}^{n} \mathcal{L}_i}{n} \tag{9}$$

D. The Configurations for the whole network

Previous work [9, 10, 32] has shown that the Transformer-based approach achieves better performance when dealing with modal general missingness, however, the model is much less effective when modal severe missingness is involved [11]. Therefore, we propose to construct sample prototypes to learn rich representations with fewer samples. Given a support set U, we assume that it contains N categories and that each category has a prototype. It is based on K tuples $(x_{m1}, y_m), \ldots, (x_{mK}, y_m)$ in U, of multimodal representations to calculate the prototype of the emotion type y_m. More precisely, the prototype representation of y_m is defined as Eq.11.

$$U = \{(x_{11}, y_1), \ldots, (x_{1K}, y_1), \ldots, (x_{NK}, y_N)\} \tag{10}$$

$$P_m(S) = \frac{1}{K} \sum_K^{i=1} L_{x_{m_i}} \tag{11}$$

Where x denotes the different modalities and y denotes the emotion category. To predict the emotion type between N ways (ways), the Euclidean distance d between the query tuple $Q = (V, L, A)$ and each prototype $P_m(S)$ is calculated, and softmax is applied to the distance vector to generate a probability distribution about the emotion category:

$$P_y(y|q) = \frac{exp\left(-d\left(L_{x_m}, P_m(U)\right)\right)}{\sum_{i=1}^{N} exp\left(-d\left(L_{x_m}, P_i(U)\right)\right)} \tag{12}$$

Compared to Bayesian networks, our method of constructing prototypes is simpler and more efficient, has better generalization for classification, and is able to accomplish the classification task when modality is completely missing, provided that a prototype representation of the classification can be generated at a higher level. We evaluated the efficiency of the algorithm on two datasets, CMU-MOSI [27] and IEMOCAP [28]. In this paper, the modality of some samples is incomplete. For instance, the text missing ratio is defined as $r = T/N$,where T is the number of samples with text modality and N is the total sample size. r quantifies the severity of missing modality. The smaller the r, the more severe the modality missing.

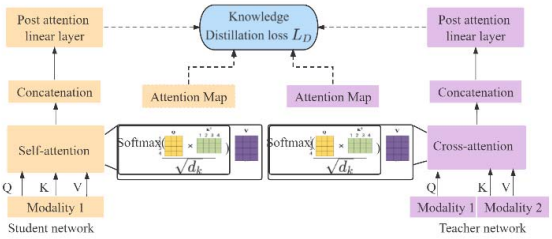

Fig. 3. Multi-level distillation network, applying two deeper representations of distillation in the attention map and post-attention linear layer.

E. Model Training

We represent the overall training objective as the final loss computed as a combination of distillation loss $\mathcal{L}'$ classification loss$\mathcal{L}_c$ (cross-entropy loss) prototype loss $\mathcal{L}_p$ (cross-entropy loss) as described in Eq. (13), which we use to train the teacher-student network. During training, we keep the training weights of the teacher network fixed and back-propagate the gradient through the student network only.

$$\mathcal{L}_T = \gamma_1 \mathcal{L}' + \gamma_2 \mathcal{L}_p + \mathcal{L}_c \tag{13}$$

$$\mathcal{L}_c = -\frac{1}{|U|} \sum_{i=1}^{|U|} y_n log \hat{y_n}$$

where λ1 and λ2 are the distillation loss function and the weighted hyperparameters of the prototype enhancement, respectively, U is the number of samples, y_n is the true label of the n-th sample, and $\hat{y_n}$is the predicted label.

IV. EXPERIMENTS

In this paper, we focus on two public multimodal sentiment analysis datasets, CMU-MOSI [27] and IEMOCAP [28], also a software dataset named SO2[34],which we briefly describe here.

A. Datasets

CMU-MOSI [27]: This is the most widely used and the largest multimodal dataset for multimodal sentiment analysis tasks. It consists of 23,454 video clips of movie reviews taken from YouTube. The dataset contains three modalities: video, audio, and text. One of the text (language) data sets is transcribed from the video and correctly identifies punctuation

marks. The sentiment labels in the dataset range from -3 (strongly negative) to 3 (strongly positive). We followed the classification setup of Yu [29].

IEMOCAP [28]: Recorded by actors from the Drama Department of the University of Southern California. IEMOCAP consists of binary random conversations in which the affective labels include 10 emotional types. We conducted experiments on this dataset and defined them as negative, neutral, and positive affective labels.

SO2 [34]: It only consists of several texts about Java APIs collected by Lin [34] at Stack Overflow in 2018, with [-2,2] indicating sentiment polarity. Of these, 9% are positive samples, 79% are neutral samples, and 12% are negative samples. We mainly use it to explore the application scenario of this approach for software engineering in section *H*.

B. Baselines and Evaluation Metrics

We compared the proposed model PAMD with state-of-the-art baselines in the recognized MSA typical datasets.

MCTN[21]: Multimodal Cyclic Transition Network (MCTN) is a method to learn a joint representation with good robustness by switching between modalities.

MMIN[9]: The model uses cascaded residual autoencoders and cyclic consistency learning to recover missing modes and a missing mode imagination network to reconstruct the impact of missing modes.

TFR-Net [8]:A Transformer-based feature reconstruction network is proposed to improve the robustness of the model to random missingness in unaligned modal sequences.

SMIL [11]: It proposes a method that can combine model flexibility and performance in the presence of severe modal deficiencies using Bayesian meta-learning networks.

TATE [10]: A Tag-Assisted Transformer Encoder (TATE) network is proposed to deal with the missing problem of uncertain modalities. A tag encoding module is designed to cover both unimodal and multimodal missing cases, thus directing the network's attention to these missing modalities.

To validate the effectiveness of our proposed model, we followed the design of Liu [29] and fully evaluated the model by calculating the classification accuracy and Macro-F1 scores.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{14}$$

$$MRE = \frac{1}{N}\sum_{i=1}^{N}\frac{|T_i \cap \hat{T}_i|}{|T_i|} \tag{15}$$

$$MPR = \frac{1}{N}\sum_{i=1}^{N}\frac{|T_i \cap \hat{T}_i|}{|\hat{T}_i|} \tag{16}$$

$$MacroF1 = \frac{2\times MPR\times MRE}{MPR+MRE} \tag{17}$$

where TP predicts the number of positive samples as positive successfully, TN predicts the number of negative samples as negative successfully, FP predicts the number of negative samples as positive incorrectly, and FN predicts the number of positive samples as negative incorrectly. For the i-th sample, T_i is the true label set, $\hat{T}_i$ is the predicted label set, and MacroF1 is the average of all sample F1 scores.

C. Overall Resultss

The results are shown in Table I. Optimal results are bolded. The principle of setting the missing rate is shown in 3.4, which is defined as the ratio of the number of missing samples to the total number of samples, and the magnitude of the missing rate reflects the severity of the modal missingness. Specifically, we set the missing ratios as (0.1, 0.2, 0.5).

TABLE I. OVERALL EXPERIMENTAL RESULTS.

Datasets	Models	0.1		0.2		0.5		AVG	
		Acc	*F1*	*Acc*	*F1*	*Acc*	*F1*	*Acc*	*F1*
MOSI	MCTN[21]	0.7987	0.5548	0.7749	0.5399	0.6811	0.4576	0.7516	0.5174
	MMIN[9]	0.8186	0.5775	0.8020	0.5538	0.7076	0.4895	0.7761	0.5403
	TFR[8]*	0.8240	-	0.7910	-	0.7110	-	0.7753	-
	TATE[10]	**0.8446**	**0.5821**	0.8125	0.5546	0.7404	0.5171	0.7992	0.5513
	SMIL[11]+	0.6070	0.5800	0.6330	0.6250	0.7131	0.7219	0.6510	0.6423
	PAMD(ours)	0.8292	0.5797	**0.8411**	**0.6321**	**0.8662**	**0.7340**	**0.8445**	**0.6486**
IEMOCAP	MCTN[21]	0.8102	0.7774	0.7827	0.7537	0.7663	0.6817	0.7864	0.7376
	MMIN[9]	0.8258	0.7885	0.8127	0.7709	0.7745	0.7058	0.8043	0.7551
	TATE[10]	**0.8509**	**0.7999**	0.8407	0.7910	0.8243	0.7443	0.8386	0.7784
	PAMD(ours)	0.8477	0.7912	**0.8492**	**0.7971**	**0.8520**	**0.8034**	**0.8496**	**0.7982**

*:The model cannot reproduce the effect
+:The binary classification performance of SMIL model on MOSI dataset
PAMD: Prototype Augmentation Multimodal Distillation ,Our Model

Experimental results on both datasets show that although the effect is not as good as the baseline TATE[10] at a missing rate of 0.1, the difference is small and still improves the performance marginally when the modal missing case is increasing (i.e., the missing rate is gradually increasing). And due to label bias in the dataset, the model as well as other baselines are decreasing, thus proving that the proposed method has good results for severe cases of modal deficiencies. Moreover, on the CMU-MOSI dataset, our model outperforms SMIL [11] by 20% in terms of accuracy under modal general missingness (e.g., r = 0.1). Even though SMIL performance tends to be better on the right side when the missingness rate is decreasing, its performance is only for the second classification (negative emotion labeling and positive emotion labeling). If the performance of triple classification is verified with this model, the advantage of our model will be even more evident, since more fine-grained types are always more difficult to predict for deep learning models.

Specifically, on the MOSI dataset, PAMD achieves comparable performance to SMIL [11] and TATE [10] when the missing rate is 0.1 (the difference in the MacroF1 metric is less than 1%). And it is much better than the end-to-end translation-based MCTN [21] model, which indicates that reconstructing the missing modal semantics minimizes the impact of missing modalities. In contrast, on the IEMOCAP dataset, PAMD achieves suboptimal results, and the difference with the optimal baseline is extremely small on both metrics, controlled at around 0.5%. When the modal deficit is severe (r = 0.5), PAMD achieves a performance improvement of more than 12% on the MOSI dataset because the prototype in the model enhances the semantics of the missing modality in the severe modal deficit case.

Overall, the average values of both PAMD metrics outperform the other baseline models at the three different

missing rates, and there is even an improvement of more than 4% in the ACC metric on MOSI and more than 1% on IEMOCAP. As a result, we can conclude that our model performs well in dealing with both general and severe mode missingness.

D. Qualitative Analysis

We further qualitatively analyzed the most advanced baseline models in the field of multimodal sentiment analysis in recent years, and the results are shown in Table II. The proposed model has something in common with other baseline models in that both consider the performance of the model in the presence of modal deficits, the difference being that we are the only study that takes into account both general deficits and severe deficits in modality.

TABLE II. QUALITATIVE ANALYSIS OF ALL BASELINES

MODEL	MMIN	TFR	TATE	SMIL	PAMD
Generally Missing	✓	✓	✓	×	✓
Severely Missing	×	×	×	✓	✓
Performance	✓	✓	✓	✓	✓
Resource Consumption	×	×	×	✓	✓
Transformer	✓	✓	✓	×	✓

Combining the experimental results in Table I, we can easily see that the transformer-based model has good performance in the absence of modal generality. In SMIL [11], for example, the model does not use Transformer to fuse multimodal information, which leads to worse performance when modal generality is missing (see Table I for details). The prototype augmentation can be useful when a modality is severely missing. In addition, we also explore the problem of resource consumption, which cannot be ignored in deep learning, and obtain a model that balances performance and resource consumption.

E. Effects of the distillation

Since different layers of the transformer carry different information, a multi-layer distillation strategy was used to minimize the required resources and inference time. Also, different knowledge is distilled into the student network by differentially constructing the teacher network to explore the impact of different knowledge sources on the model.

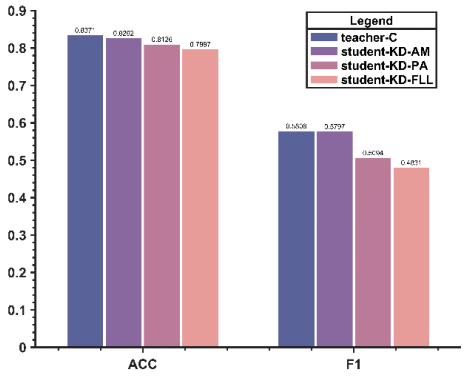

Fig. 4. Transformer different layer distillation effects. teacher-C denotes the complete teacher network with full modal training, AM,PA,FLL denote Transformer different layers in the student network respectively.

In Fig. 4, we fix the distillation knowledge sources and set distillation in different layers, including Attention Map layer, Post Attention layer, and Final Linear layer. The experimental results show that the accuracy in the optimal case is 82.92%, which outperforms the unimodal performance (third row in Table III. In addition, we find that the performance improvement occurs at the lower layer of the Transformer, i.e., the Attention map layer. We speculate that this may be due to the fact that the higher layers do not provide the student network with enough information to the Transformer. We further compared the number of parameters for the best-performing student network with the full teacher network, 1.112M and 1.457M, respectively, with a decrease in the number of parameters and an approximately 17ms reduction in inference time.

TABLE III. EFFECT OF DIFFERENT SOURCES EON DISTILLATION

MODEL	Setting	ACC	F1
teacher	C	0.8371	0.5808
student	Without KD	0.7315	0.4426
	KD from complete teacher	0.8189	0.5683
	KD from video and language	**0.8292**	**0.5797**
	KD from audio and language	0.8224	0.5740
	KD from audio and video	0.8257	0.5765
	KD from one branches of teacher	*A 0.8004* *V 0.8129* *L 0.8066*	*A 0.5573* *V 0.5637* *L 0.5618*
C:Full modal trained network of teachers *A:AUDIO BRANCH V:VIDEO BRANCH L:LANGUAGE BRANCH*			

In addition, we explored the effect of different knowledge sources on the model by differentially constructing the teacher network to distill different knowledge into the student network. The experimental results are shown in Table III. When the knowledge sources are videos and texts from the teacher network, the student network performs better than the distillation results branching from the complete teacher network, and the optimal student network inference time is much smaller than the complete teacher network. We conjecture that the reason is that the text is transcribed from audio and there is a large information overlap, causing the model to force to learn the shared knowledge between the two modalities. This reduces the discrepancy and decreases the accuracy. Furthermore, when the distillation source is unimodal, the video modality outperforms the other modality on both metrics, which is consistent with our subjective understanding (unimodality is limited in the information it can provide, while video is the unimodal form that carries relatively more information).

F. Effects of the prototype

To explore the effect of the prototype setting in PAMD, we evaluated our model using different missing rate settings. The experimental results show (Fig. 5) that not only are the accuracy and F1 much lower when the prototype network is removed than when the full model is in place, but also the performance degrades when the modal missing condition is continuously aggravated, just like with the other baseline models. Thus, it proves the necessity and effectiveness of the prototype network in coping with severe modal deficiencies.

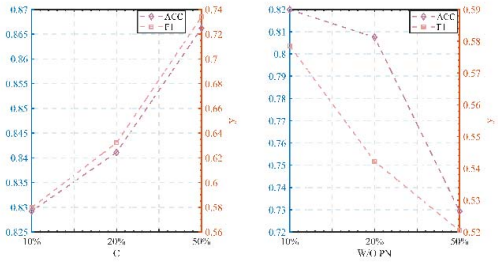

Fig. 5. Performance of different model setups with different missing rates The left figure represents the complete distillation network, and the right figure indicates the distillation network after removing the prototype enhancement.

G. Ablation Study

Finally, we conducted ablation experiments on the CMU-MOSI dataset. After fixing the missing rate, we tested the performance of the model after removing each module of the model separately, and the results are shown in Table 4. From Table IV, we can see that removing any module from the PAMD leads to a decrease in model performance. Specifically, the removal of the cross-modal intra-attention module had the largest impact on model performance, resulting in a decrease in accuracy of about 13% due to the severing of potential connections between different modalities and the regression of the model's ability to learn back to its original state. In addition, the teacher network had the second largest impact on model performance, indicating that our distillation design is essential. The prototype augmentation design was originally intended to cope with the severe absence of modalities, so when modalities were missing in general (10%), the impact was not significant.

TABLE IV. EFFECT OF DIFFERENT SOURCES EON DISTILLATION

MODEL	Metrics	
	ACC	*F1*
PAMD(w/o p)	0.8199	0.5786
PAMD(w/o t)	0.7315	0.4426
PAMD(w/o a)	0.7034	0.4177
PAMD	**0.8292**	**0.5797**

H. Case Study

In a practical software engineering scenario, we are more likely to face a situation where a modality is completely missing (r = 100%). For example, it is almost impossible to select data from three modalities—text, image, and video—at the same time when posting dynamics in technical communities. Therefore, in order to more intuitively demonstrate the effectiveness of our model for sentiment analysis in the software engineering domain, we set up different modality-missing scenarios and selected software engineering domain data for testing, and the results are shown in Fig.6.

In Example A, we use the SO2 dataset annotated by Lin [34], which contains 1500 sentences about APIs extracted from Stack Overflow. Each sentence is annotated with sentiment strength by two annotators, where -2 represents strong negative, -1 represents weak negative, 0 represents neutral, 1 represents weak positive, and 2 represents strong positive. We treat them as missing video and audio modalities, and the results of all three models have a small gap with the true labels, thus demonstrating the good performance of the model on unimodality. It is worth mentioning that the difference between the PAMD of our model and the real label is negligible, only 0.04, thus proving the good performance of our model.

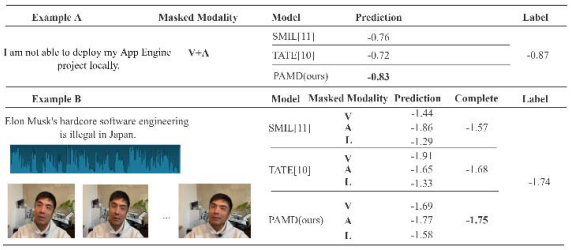

Fig. 6. Case study in the field of software engineering, where we randomly lost different modalities in the test

In Example B, we selected user-published dynamic data from CSDN, a technical community related to software engineering, and weighted the average as the true sentiment label value after annotation by five people with software engineering-related backgrounds according to the annotation principle in [34]. We conducted correlation experiments on different models after randomly discarding different modalities. The third column (Masked Modality)indicates the lost modalities, the fourth column (Predication) indicates the corresponding predicted values for the different models, and the fifth column (Complete)indicates the performance of the full model in the full modality.

Overall, our model predicts the smallest gap between the sentiment label values and the true labels. In addition, the experimental results were the worst among the three models after discarding textual data, thus proving that textual information contains more semantics and dominates in multimodal sentiment analysis in software engineering. In contrast, when the discarded modality was audio, all three models achieved the smallest gap with the real label, i.e., the optimal result, due to the fact that video and text data always carry more information. TATE [12] outperforms SMIL [11] because it uses the entire modality to pre-train the guidance network and the forward JS divergence loss can be used as a good supervision. In contrast, our model PAMD uses a transformer-based distillation network with prototype enhancement and achieves better results in the face of severe modal deficiencies.

V. CONCLUSION AND FUTURE WORK

In this paper, we design a Transformer-based teacher-student network (PAMD) for the unavoidable modal missing case in MSA tasks in software engineering, and the model is able to handle different modal missing cases flexibly. The core of the model employs a multi-level and multi-origin distillation strategy to minimize the required resources and inference time and a prototype enhancement to ensure the performance of the model when a modality is severely missing.

We find that the network performance is not positively related to the number of modalities from distillation sources, which provides inspiration for exploring lightweight and efficient models. The experiments use software engineering data from the same platform (e.g., stack overflow) as training and test sets, and will subsequently explore the effect of cross-platform settings for sentiment analysis tasks in software

engineering, thus providing help for customized sentiment analysis in software engineering.

ACKNOWLEDGMENT

his work was supported by the National Natural Science Foundation of China under Grant No. 61862063, 61502413, 61262025; the National Social Science Foundation of China under Grant No. 18BJL104; the Science Foundation of Young and Middle-aged Academic and Technical Leaders of Yunnan under Grant No. 202205AC160040; the Science Foundation of Yunnan Jinzhi Expert Workstation under Grant No. 202205AF150006; the Open Foundation of Yunnan Key Laboratory of Software Engineering under Grant No. 2020SE301; the Science Foundation of "Knowledge-driven intelligent software engineering innovation team".

REFERENCES

[1] Huq SF, Sadiq AZ, Sakib K. Is developer sentiment related to software bugs: An exploratory study on GitHub commits. In: Proc. of the 27th IEEE Int'l Conf. on Software Analysis, Evolution and Reengineering. London: IEEE, 2020. 527–531.

[2] Chen Zhenpeng,Yao Huihan,Cao Yanbin,Liu Xuanzhe,Mei Hong. Research on sentiment analysis techniques for software engineering[J/OL]. Journal of Software:1-13[2022-11-23].

[3] Matthias Springstein, Eric Müller-Budack, and Ralph Ewerth. 2021. QuTI! Quantifying Text-Image Consistency in Multimodal Documents. In Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval. 2575–2579.

[4] Hazarika D, Zimmermann R, Poria S. Misa: Modality-invariant and-specific representations for multimodal sentiment analysis[C]//Proceedings of the 28th ACM international conference on multimedia. 2020: 1122-1131.

[5] Jie Xu, Zhoujun Li, Feiran Huang, Chaozhuo Li, and S Yu Philip. 2020. Social image sentiment analysis by exploiting multimodal content and heterogeneous relations. IEEE Transactions on Industrial Informatics 17, 4 (2020), 2974–2982.

[6] Liu X, Zhu X, Li M, et al. Multiple kernel $ k $ k-means with incomplete kernels[J]. IEEE transactions on pattern analysis and machine intelligence, 2019, 42(5): 1191-1204.

[7] Srinivas Parthasarathy and Shiva Sundaram. 2020. Training strategies to handle missing modalities for audio-visual expression recognition. In Companion Publication ofthe 2020 International Conference on Multimodal Interaction. 400–404.

[8] Ziqi Yuan, Wei Li, Hua Xu, and Wenmeng Yu. 2021. Transformer-based feature reconstruction network for robust multimodal sentiment analysis. In Proceedings ofthe 29th ACM International Conference on Multimedia. 4400–4407.

[9] Jinming Zhao, Ruichen Li, and Qin Jin. 2021. Missing Modality Imagination Network for Emotion Recognition with Uncertain Missing Modalities. In Proceedings ofthe 59th Annual Meeting ofthe Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing. 2608–2618.

[10] Zeng J, Liu T, Zhou J. Tag-Augmentation Multimodal Sentiment Analysis under Uncertain Missing Modalities[J]. arXiv preprint arXiv:2204.13707, 2022.

[11] Ma M, Ren J, Zhao L, et al. SMIL: Multimodal learning with severely missing modality[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2021, 35(3): 2302-2310.

[12] Jennifer Williams, Steven Kleinegesse, Ramona Comanescu, and Oana Radu. 2018. Recognizing Emotions in Video Using Multimodal DNN Feature Fusion. In Proceedings ofGrand Challenge andWorkshop on Human Multimodal Language (Challenge-HML). 11–19.

[13] Amir Zadeh, Paul Pu Liang, Navonil Mazumder, Soujanya Poria, Erik Cambria, and Louis-Philippe Morency. 2018. Memory Fusion Network for Multi-view Sequential Learning.. In AAAI. 5634–5641.

[14] Sijie Mai, Songlong Xing, and Haifeng Hu. 2021. Analyzing multimodal sentiment via acoustic-and visual-LSTM with channel-aware temporal convolution network. IEEE/ACMTransactions on Audio, Speech, andLanguage Processing 29 (2021), 1424– 1437.

[15] Navonil Majumder, Devamanyu Hazarika, Alexander Gelbukh, Erik Cambria, and Soujanya Poria. 2018. Multimodal sentiment analysis using hierarchical fusion with context modeling. Knowledge-based Systems 161 (2018), 124–133.

[16] S. Poria, D. Hazarika, N. Majumder, and R. Mihalcea. Beneath the tip of the iceberg: Current challenges and new directions in sentiment analysis research. IEEE Transactions on Affective Computing, 2020.

[17] Srinivas Parthasarathy and Shiva Sundaram. 2020. Training strategies to handle missing modalities for audio-visual expression recognition. In Companion Publication ofthe 2020 International Conference on Multimodal Interaction, pages 400–404.

[18] Luan Tran, Xiaoming Liu, Jiayu Zhou, and Rong Jin. 2017. Missing modalities imputation via cascaded residual autoencoder. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 1405–1414.

[19] Pierre Baldi. 2012. Autoencoders, unsupervised learning, and deep architectures. In Proceedings ofICML Workshop on Unsupervised and Transfer Learning. 37–49.

[20] Lei Cai, Zhengyang Wang, Hongyang Gao, Dinggang Shen, and Shuiwang Ji. 2018. Deep adversarial learning for multi-modality missing data completion. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pages 1158–1166.

[21] [Hai Pham, Paul Pu Liang, Thomas Manzini, Louis-Philippe Morency, and Barnabás Póczos. 2019. Found in translation: Learning robust joint representations by cyclic translations between modalities. In Proceedings ofthe AAAIConference on Artificial Intelligence, Vol. 33. 6892–6899.

[22] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network, 2015.

[23] N. Kitaev, Łukasz Kaiser, and A. Levskaya. Reformer: The efficient transformer, 2020.

[24] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. A simple framework for contrastive learning of visual representations, 2020.

[25] W. Dai, S. Cahyawijaya, Z. Liu, and P. Fung. Multimodal end-to-end sparse model for emotion recognition. arXiv preprint arXiv:2103.09666, 2021.

[26] Y.-H. H. Tsai, S. Bai, P. P. Liang, J. Z. Kolter, L.-P. Morency, and R. Salakhutdinov. Multimodal transformer for unaligned multimodal language sequences. In Proceedings ofthe conference. Association for

[27] Amir Zadeh, Rowan Zellers, Eli Pincus, and Louis-Philippe Morency. 2016. Multimodal sentiment intensity analysis in videos: Facial gestures and verbal messages. IEEE Intelligent Systems 31, 6 (2016), 82–88.

[28] Carlos Busso, Murtaza Bulut, Chi-Chun Lee, Abe Kazemzadeh, Emily Mower,

[29] Wenmeng Yu, Hua Xu, Ziqi Yuan, and Jiele Wu. 2021. Learning modality-specific representations with self-supervised multi-task learning for multimodal sentiment analysis. In Proceedings ofthe AAAI Conference on Artificial Intelligence. 10790–10797..

[30] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. 2017. Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks. In 2017 IEEE International Conference on Computer Vision (ICCV). 2242–2251.

[31] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All You Need. In Proceedings ofthe 31st International Conference on Neural Information Processing Systems, Vol. 30. 5998–6008.

[32] Zhang Q, Shi L, Liu P, et al. ICDN: integrating consistency and difference networks by transformer for multimodal sentiment analysis[J]. Applied Intelligence, 2022: 1-14.

[33] Gong Z, Gao C, Wang Y, et al. Source Code Summarization with Structural Relative Position Guided Transformer [C]//2022 IEEE International Conference on Software Analysis, Evolution and Reengineering (SANER). 2022.

[34] Lin B, Zampetti F, Bavota G, Di Penta M, Lanza M, Oliveto R. Sentiment analysis for software engineering: How far can we go? In: Proc. of the 40th Int'l Conf. on Software Engineering. Gothenburg: ACM, 2018. 94–104.

[35] Ortu M, Adams B, Destefanis G, Tourani P, Marchesi M, Tonelli R. Are bullies more productive? Empirical study of affectiveness vs. issue fixing time. In: Proc. of the 12th IEEE/ACM Working Conf. on Mining Software Repositories. Florence: IEEE, 2015. 303–313.