Tag-assisted Multimodal Sentiment Analysis under Uncertain Missing Modalities

Jiandian Zeng
State Key Laboratory of IoT for Smart City, University of Macau
Macau, China
y87470@um.edu.mo

Tianyi Liu
Shanghai Jiao Tong University
Shanghai, China
liutianyi@sjtu.edu.cn

Jiantao Zhou
State Key Laboratory of IoT for Smart City, University of Macau
Macau, China
jtzhou@um.edu.mo

ABSTRACT
Multimodal sentiment analysis has been studied under the assumption that all modalities are available. However, such a strong assumption does not always hold in practice, and most of multimodal fusion models may fail when partial modalities are missing. Several works have addressed the missing modality problem; but most of them only considered the single modality missing case, and ignored the practically more general cases of multiple modalities missing. To this end, in this paper, we propose a Tag-Assisted Transformer Encoder (TATE) network to handle the problem of missing uncertain modalities. Specifically, we design a tag encoding module to cover both the single modality and multiple modalities missing cases, so as to guide the network’s attention to those missing modalities. Besides, we adopt a new space projection pattern to align common vectors. Then, a Transformer encoder-decoder network is utilized to learn the missing modality features. At last, the outputs of the Transformer encoder are used for the final sentiment classification. Extensive experiments are conducted on CMU-MOSI and IEMOCAP datasets, showing that our method can achieve significant improvements compared with several baselines.

CCS CONCEPTS
• Information System → Multimodal Sentiment Analysis.

KEYWORDS
Multimodal Sentiment Analysis, Missing Modality, Joint Representation

1 INTRODUCTION
Nowadays, sentiment analysis has attracted intensive interest in extracting human’s emotion and opinion [36, 47], among which multimodal sentiment analysis is becoming an especially popular research direction with the massive amounts of online content. Besides, it has been shown that combining different modalities can learn complementary features, resulting in better joint multimodal representations [30, 31]. Most prior works on multimodal fusion [6, 38, 39] assumed that all modalities are always available when training and testing. However, in real life, we often encounter scenarios that partial modalities could be missing. For example, as shown in Fig. 1, the visual features may be blocked due to the non-coverage of camera; the acoustic information may be unavailable due to the enormous ambient noise; and the textual information may be absent due to the privacy issue. Therefore, how to handle missing modalities is emerging as a hot topic in the multimodal area.

Previous works [23, 25, 29] simply discarded missing modalities or utilized matrix completion methods to impute missing modalities, and somewhat degraded overall performance. Zhao et al. [29] completed the kernel matrices of the modality using the common instances in different modalities. In [25], the visual modality was ablated when training with missing data. Owing to the strong learning ability of deep learning, recent works have employed neural networks to learn latent relationships among available modalities. To tackle the missing modality problem, Tran et al. [33] first identified the general problem of missing modality in multimodal data, and proposed a Cascaded Residual Auto-encoder (CRA) network to learn complex relationship from different modalities. More recently, Zhao et al. [49] adopted cycle consistency learning with CRA to recover missing modalities. Yuan et al. [44] designed a Transformer-based feature reconstruction network to guide the
extractor in obtaining the semantics of missing modality features. However, most of the above works all assumed that there is only one missing modality, and ignored the practically more general cases of multiple modalities missing. That is, they require training a new model to fit each missing modality case, which is both costly and inconvenient. In reality, the pattern of missing modalities could be uncertain, e.g., one or two modalities are randomly absent. To tackle the above issues, two challenges should be addressed: 1) will the model still work when multiple modalities are absent? and 2) how to learn robust joint representations when the missing modalities are uncertain?

In this paper, we propose a Tail-Assisted Transformer Encoder (TATE) network to learn complementary features among modalities. For the first challenge, we design a tag encoding module to mark missing modalities, aiming to direct the network’s attention to absent modalities. As will be shown later, the attached tag not only can cover both the single modality and multiple modalities absent situations, but also can assist in joint representation learning. For the second challenge, we first adopt the Transformer [34] as the extractor to capture intra-modal features, and then apply a two-by-two projection pattern to map them into a common space. After that, the pre-trained network trained with full modalities is utilized to supervise the encoded vectors. At last, the outputs generated by a Transformer encoder are fed into a classifier for sentiment prediction. Our contributions are summarized as follows:

- We propose the TATE network to handle the multiple modalities missing problem for multimodal sentiment analysis. The code is publicly available\(^1\).
- We design a tag encoding module to cover both the single modality and multiple modalities absent situations, and adopt a new common space projection module to learn joint representations.
- Our proposed model TATE achieves significant improvements compared with several benchmarks on CMU-MOSI and IEMOCAP datasets, validating the superiority of our model.

2 RELATED WORKS

In this section, we first introduce the concept of multimodal sentiment analysis, and then review the related methods of handling missing modalities.

2.1 Multimodal Sentiment Analysis

As a core branch of sentiment analysis [11, 50], multimodal sentiment analysis has attracted significant attention in recent years [21, 27, 32, 41]. Compared to a single modality case, multimodal sentiment analysis is more challenging due to the complexity of handling and analyzing data from different modalities.

To learn joint representations of multimodal, three multimodal fusion strategies are applied: 1) early fusion directly combines features of different modalities before the classification. Majumder et al. [22] proposed a hierarchical fusion strategy to fuse acoustic, visual and textual modalities, and proved the effectiveness of two-by-two fusion pattern; 2) late fusion adopts the average score of each modality as the final weights. Guo et al. [13] adopted an online early-late fusion scheme to explore complementary relationship for the sign language recognition, where late fusion further aggregated features combined by the early fusion; and 3) intermediate fusion utilizes a shared layer to fuse features. Xu et al. [40] constructed the decomposition and relation networks to represent the commonality and discrepancy among modalities. Hazarika et al. [14] designed a multimodal learning framework that can learn modality-invariant and modality-specific representations by projecting each modality into two distinct sub-spaces. However, few of the above multimodal fusion models can handle the cases when partial modalities are missing.

2.2 Missing Modalities Methods

In recent years, many works focused on handling the missing modality problem, and they can be generally categorized into two groups: 1) generative methods [2, 10, 28, 33, 48], and 2) joint learning methods [26, 37, 44, 49].

Generative methods learn to generate new data with similar distributions to obey the distribution of the observed data. With the ability to learn latent representations, the auto-encoder (AE) [2] is widely used. Vincent et al. [35] extracted features with AE based on the idea of making the learned representations robust to partial corruption of the input data. Kingma et al. [18] designed a Variational Auto-Encoder (VAE) to infer and learn features with simple ancestral sampling. Besides, inspired by the residual connection network [15], Tran et al. [53] proposed a Cascaded Residual Auto-encoder (CRA) to impute data with missing modality, which combined a series of residual AEs into a cascaded architecture to learn relationships among different modalities. As for the Generative Adversarial Networks (GAN) [12], Shang et al. [28] treated each view as a separate domain, and identified domain-to-domain mappings via a GAN using randomly-sampled data from each view. Besides, the domain mapping technique is also considered to impute missing data. Cai et al. [5] formulated the missing modality problem as a conditional image generation task, and designed a 3D encoder-decoder network to capture modality relations. They also incorporated the available category information during training to enhance the robustness of the model. Moreover, Zhao et al. [48] developed a cross partial multi-view network to model complex correlations among different views, where multiple discriminators are used to generate missing data.

Joint learning methods try to learn joint representations based on the relations among different modalities [1, 16, 26]. Based on the idea that the cycle consistency loss can retain maximal information from all modalities, Pham et al. [26] investigated learning robust representations via cyclic translations from source to target modalities. Zhao et al. [49] also applied cycle consistency learning for missing modality imputation, where the CRA-based cross-modality imagination module is designed based on paired multimodal data. More recently, Yuan et al. [44] utilized the Transformer to extract intra-modal and inter-modal relations, and designed a Transformer-based feature reconstruction network to reproduce the semantics of missing modality.

However, most of the above works can only handle the scenarios of missing a single modality, and cannot satisfactorily deal with multiple modalities missing cases since they need to train a new

\(^1\)https://github.com/JaydenZeng/TATE
model for each case. As will be clear soon, our works differs the
above works in several ways: 1) a tag encoding module is designed
to cover all uncertain missing cases; and 2) a new mapping method
is applied to learn joint representations in the common space pro-
jection module.

3 METHODOLOGY
In this section, we first give the problem definition and associated
notations. Then, we present the overall workflow of the proposed
architecture and the detailed modules.

3.1 Problem Definition and Notations
Given a multimodal video segment that contains three modalities:
\( S = \{X_v, X_a, X_t\} \), where \( X_v \), \( X_a \) and \( X_t \) denote visual, acoustic and
textual modalities respectively. Without loss of generality, we use
\( X'_m \) to represent the missing modality, where \( m \in \{v, a, t\} \). For
instance, assuming that the visual modality and acoustic modality
are absent, and the multimodal representation can be denoted as
\( [X'_v, X'_a, X_t] \). The primary task is to classify the overall sentiment
(positive, neutral, or negative) under uncertain missing modalities.

3.2 Overall Framework
As can be seen in Fig. 2, the main workflow is as follows: for a
given video segment, assuming that the visual modality and acous-
tic modality are missing, we first mask these missing modalities
as 0, and then extract the remaining raw features. Afterwards, the
masked multimodal representation goes through two branches:
1) one is encoded by a pre-trained model, which is trained with
all full modality data, and 2) another goes through the tag encod-
ing module and the common space projection module to acquire
aligned feature vectors. Then, the updated representations are pro-
cessed by a Transformer encoder, and we calculate the forward
similarity loss between the pre-trained vectors and the encoder
outputs. Meanwhile, the encoded outputs are fed into a classifier
for the sentiment prediction. At last, we compute the backward
reconstruction loss and the tag recovery loss to supervise the joint
representation learning. Each module will be introduced clearly in
following sub-sections.

3.3 Multi-Head Attention
Transformer \([34]\) not only plays a great role in the Natural Language
Processing (NLP) community, but also shows excellent representa-
tional capabilities in other areas, such as Computer Vision (CV) \([7]\).
Instead of using an RNN based structure to capture the sequential
information, we employ the Transformer to generate the contextu-
ral representation of each modality respectively, where the key
component of multi-head dot-product attention can be formalized
as follows:

\[
\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d}})V,
\]

where \( Q, K \) and \( V \) are the query, the key, and the value respectively,
and \( d \) is the dimension of the input.

Instead of utilizing the single attention, the multi-head attention
is applied to obtain more information from different semantic
spaces:

\[
E_M = \text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \text{head}_2, ..., \text{head}_h)W^o,
\]
where \( W^e \in \mathbb{R}^{d \times d} \) is a weight matrix, \( h \) is the head number. Given the input \( E \), the \( i \)-th head is calculated as follow:

\[
\text{head}_i = \text{Attention}(E W^Q_i, E W^K_i, E W^V_i)
\]

(3)

where \( W^Q_i \in \mathbb{R}^{d \times d}, W^K_i \in \mathbb{R}^{d \times d} \) and \( W^V_i \in \mathbb{R}^{d \times d} \) are the \( i \)-th weight matrices of the query, the key and the value.

Therefore, the updated modalities representations can be formulated as follows:

\[
\begin{align*}
E_o &= \text{MultiHead}(X'_o, X'_o, X'_o), \\
E_a &= \text{MultiHead}(X'_a, X'_a, X'_a), \\
E_t &= \text{MultiHead}(X_t, X_t, X_t).
\end{align*}
\]

(4)

### 3.4 Tag Encoding

To specify uncertain missing modalities, we employ a tag encoding module to mark them, and direct network’s attention to these disabled modalities. In our settings, we adopt 4 digits ("0" or "1") to label missing modalities. If partial modalities of the input are missing, we set the first digit as "0", otherwise "1". Besides, the last three digits are used to mark the corresponding visual, acoustic and textual modalities. As can be seen in Fig. 3, we give two examples about modality tags: in Fig. 3a, the acoustic modality is disabled modalities. In our settings, we adopt 4 digits ("0" or "1") to label missing modalities. If partial modalities of the input are missing, we set the first digit as "0", otherwise "1". Besides, the last three digits are used to mark the corresponding visual, acoustic and textual modalities. As can be seen in Fig. 3, we give two examples about modality tags: in Fig. 3a, the acoustic modality is missing, and the tag is set as "0010"; for multiple modalities missing cases (Fig. 3b), we set the tag as "0110" to mark visual and acoustic modalities. The benefits are twofold: 1) the tag encoding module can cover both single and multiple modalities missing conditions; and 2) the encoded tags can complementarily assist in the learning of the joint representations. To simplify mathematical expression, we denote all tags as \( E_{tag} \).

### 3.5 Common Space Projection

After the tag encoding module, we now project three modalities into the common space. Previous works [14, 40] that directly utilized simple feed-forward neural layers with same parameters for the projection, which may be failed when there are more than two modalities. To tackle the issue, we adopt a two-by-two projection pattern to acquire a more general space. As shown in Fig. 4, for each single modality, we first obtain the self-related common space based on the following linear transformation:

\[
\begin{align*}
C_o &= [W_{oa} E_o || W_{ot} E_o], \\
C_a &= [W_{oa} E_a || W_{at} E_a], \\
C_t &= [W_{ot} E_t || W_{at} E_t],
\end{align*}
\]

(5)

where \( W_{oa}, W_{ot} \) and \( W_{at} \) are all weight matrices, and \( || \) denotes the vertical concatenating operation. Then, we concatenate all common vectors and the encoded tag to eventually obtain the common joint representations \( E_{all} \):

\[
E_{all} = [C_o || C_a] || C_t || E_{tag}.
\]

(6)

### 3.6 Transformer Encoder-Decoder

To effectively model the long-term dependency of the intra-modal and the inter-modal information, we employ one sub-layer in Transformer [34] to manage the information flow. As illustrated in Section 3.3, the encoded outputs \( E_{out} \) can be accessed by the multi-head attention and feed-forward networks:

\[
\begin{align*}
E_{out} &= \text{MultiHead}(E_{all}, E_{all}, E_{all}), \\
E_{out} &= \text{Relu}(E_{out} W^c_1 + b^1_c) W^c_2 + b^2_c.
\end{align*}
\]

(7)

where the query, the key, and the value are the same input \( E_{all} \); \( W^c_1 \) and \( W^c_2 \) are two weight matrices, \( b^1_c \) and \( b^2_c \) are two learnable biases.

Similarly, the decoded outputs \( D_{out} \) are formulated as follows:

\[
D_{out} = \text{MultiHead}(E_{out}, E_{out}, E_{out}), \\
D_{out} &= \text{Relu}(D_{out} W^d_1 + b^1_d) W^d_2 + b^2_d,
\]

(8)

where \( W^d_1, W^d_2, b^1_d, \) and \( b^2_d \) are parameters.

### 3.7 Training Objective

The overall training objective \( \mathcal{L}_{total} \) is expressed as:

\[
\mathcal{L}_{total} = \mathcal{L}_{cls} + \lambda_1 \mathcal{L}_{forward} + \lambda_2 \mathcal{L}_{backward} + \lambda_3 \mathcal{L}_{tag},
\]

(9)

where \( \mathcal{L}_{cls} \) is the classification loss, \( \mathcal{L}_{forward} \) is the forward differential loss, \( \mathcal{L}_{backward} \) is the backward reconstruction loss, \( \mathcal{L}_{tag} \) is the tag recovery loss, and \( \lambda_1, \lambda_2, \) and \( \lambda_3 \) are the corresponding weights. We now introduce the loss terms in details.
3.7.1 Forward Differential Loss ($L_{\text{forward}}$). As illustrated in Fig. 2, the forward loss is calculated by the difference between the pre-trained output ($E_{\text{pre}}$) and the Transformer encoder output ($E_{\text{out}}$). Similar to [49], the pre-trained model is trained with full modality data, where features from three modalities are directly concatenated for classification. Thus, we employ the differential loss to guide the learning process for missing modalities. Specifically, the Kullback Leibler (KL) divergence loss is used:

$$D_{KL}(p||q) = \sum_{i=1}^{N} p(x_i) \cdot \frac{p(x_i)}{q(x_i)}$$

where $p$ and $q$ are two probability distributions. Since KL divergence is asymmetric, we adopt the Jensen-Shannon (JS) divergence loss instead:

$$L_{\text{forward}} = JS(E_{\text{out}}||E_{\text{pre}}) = \frac{1}{2}(D_{KL}(E_{\text{out}}||E_{\text{pre}}) + D_{KL}(E_{\text{pre}}||E_{\text{out}})).$$

3.7.2 Backward Reconstruction Loss ($L_{\text{backward}}$). For the backward loss, we aim to supervise the joint common vector reconstruction. Therefore, similar to the forward differential loss, we calculate the JS divergence loss between the Transformer decoder output ($D_{\text{out}}$) and the updated common joint representations ($E_{\text{all}}$):

$$L_{\text{backward}} = JS(D_{\text{out}}||E_{\text{all}}) = \frac{1}{2}(D_{KL}(D_{\text{out}}||E_{\text{all}}) + D_{KL}(E_{\text{all}}||D_{\text{out}})).$$

3.7.3 Tag Recovery Loss ($L_{\text{tag}}$). In our settings, the tag is attached to mark missing modalities, and we expect our network can pay more attention to them. To better guide the reconstruction of the attached tag, we design a tag recovery loss to direct the process. The reason why we choose the Mean Absolute Error (MAE) loss is that MAE is less sensitive to outliers with the absolute function. Thus, MAE is adopted to calculate the loss between $E_{\text{tag}}$ and the last four digits of $D_{\text{out}}$:

$$L_{\text{tag}} = \frac{1}{N} \sum_{i=1}^{N} |E_{\text{tag}} - D_{\text{tag}}|,$$

$$D_{\text{tag}} = \text{Sigmoid}(D_{\text{out}} \cdot [-4 :]) .$$

3.7.4 Classification Loss ($L_{\text{cls}}$). For the final classification module, we feed $E_{\text{out}}$ into a fully connected network with the softmax activation function:

$$P_c = \text{softmax}(W_c E_{\text{out}} + b_c),$$

where $W_c$ and $b_c$ are the learned weights and bias. In detail, we employ the standard cross-entropy loss for this task, that is:

$$L_{\text{cls}} = -\frac{1}{N} \sum_{n=1}^{N} y_n \log \hat{y}_n,$$

where $N$ is the number of samples, $y_n$ is the true label of the $n$-th sample, and $\hat{y}_n$ is the predicted label.

### 4 EXPERIMENTS

All experiments are carried out on a Linux server (Ubuntu 18.04.1) with an Intel(R) Xeon(R) Gold 5120 CPU, 8 Nvidia 2080TI GPUs and 128G RAM. Datasets and experimental settings are described as follows:

**Datasets:** We conduct several experiments on CMU-MOSI [45] and IEMOCAP [4] datasets. Both datasets are multimodal benchmarks for sentiment recognition, including visual, textual, and acoustic modalities. For the CMU-MOSI dataset, it contains 2199 segments from 93 opinion videos on YouTube. The label of each sample is annotated with a sentiment score in [-3, 3]. Following Yu et al. [43], we transform the score into negative, neutral and positive labels. For the IEMOCAP dataset, it contains 5 sessions, and each session contains about 30 videos, where each video contains at least 24 utterances. The annotated labels are: neutral, frustration, anger, sad, happy, excited, surprise, fear, disappointing, and other. Specifically, we report three-classes (negative: [-3,0], neutral:[0], positive: (0,3)) results on CMU-MOSI, and two-classes (negative:frustration, angry, sad, fear, disappointing), positive: [happy, excited]) on IEMOCAP.

**Parameters:** Following standard methods, we tune our model using five-fold validation and grid-searching on the training set. The learning rate $lr$ is selected from $[0.1, 0.001, 0.0005, 0.0001]$, the batch size $b \in \{32, 64, 128\}$, and the hidden size $d \in \{64, 128, 300, 768\}$. Adam [17] is adopted to minimize the total loss given in Eq. (9). The epoch number is 20, the batch size is 32, the loss weight is set to 0.1, and the parameters are summarized in Table 1.

**Evaluation Metric:** Accuracy and Macro – F1 are used to measure the performance of the models, which are defined as follows:

$$\text{Accuracy} = \frac{T_{\text{true}}}{N},$$

$$F_1 = \frac{2PR}{P + R},$$

where $T_{\text{true}}$ is the number of correctly predicted samples, $N$ is the total number of samples, $P$ is the positive predictive value, and $R$ is the recall value.

#### 4.1 Feature Extraction

**Visual Representations:** The CMU-MOSI [45] and IEMOCAP [4] datasets mainly consist of human conversations, where visual features are mainly composed of human faces. Following [42, 46], we also adopt OpenFace2.0 toolkit [3] to extract facial features.
### Table 2: Results of all baselines of missing a single modality, where the best results are in bold.

| Datasets | Models | 0  | 0.1  | 0.2  | 0.3  | 0.4  | 0.5  |
|-----------|--------|----|------|------|------|------|------|
|           |        | M-F1 | ACC  | M-F1 | ACC  | M-F1 | ACC  |
| CMU-MOSI  | AE     | 56.78 | 79.69 | 54.07 | 79.17 | 53.40 | 78.01 |
|           | CRA    | 56.85 | 79.73 | 54.37 | 79.87 | 53.99 | 77.49 |
|           | MCTN   | 57.32 | 79.75 | 55.48 | 79.87 | 53.99 | 77.49 |
|           | TransM | 57.84 | 80.21 | 57.53 | 79.69 | 55.21 | 78.42 |
|           | MMIN   | 60.41 | 82.29 | 57.75 | 81.86 | 55.38 | 80.20 |
|           | Ours   | 58.32 | 84.90 | 58.21 | 84.46 | 55.46 | 81.25 |
| IEMOCAP   | AE     | 76.15 | 82.09 | 75.24 | 80.26 | 75.02 | 78.01 |
|           | CRA    | 77.05 | 82.13 | 75.95 | 80.97 | 75.13 | 78.09 |
|           | MCTN   | 78.57 | 82.27 | 77.74 | 81.02 | 75.37 | 78.27 |
|           | TransM | 79.57 | 82.64 | 78.03 | 81.86 | 76.33 | 80.43 |
|           | MMIN   | 80.83 | 83.43 | 78.85 | 82.58 | 77.09 | 81.27 |
|           | Ours   | 81.15 | 85.39 | 79.99 | 85.09 | 79.10 | 84.07 |

### Table 3: Results of all baselines of missing multiple modalities, where the best results are in bold.

| Datasets | Models | 0  | 0.1  | 0.2  | 0.3  | 0.4  | 0.5  |
|-----------|--------|----|------|------|------|------|------|
|           |        | M-F1 | ACC  | M-F1 | ACC  | M-F1 | ACC  |
| CMU-MOSI  | AE     | 56.78 | 79.69 | 52.80 | 75.65 | 50.84 | 74.18 |
|           | CRA    | 56.81 | 79.72 | 52.85 | 75.68 | 51.02 | 74.73 |
|           | MCTN   | 56.85 | 79.73 | 52.97 | 75.89 | 51.75 | 74.16 |
|           | TransM | 57.84 | 80.21 | 53.49 | 77.08 | 51.97 | 74.24 |
|           | MMIN   | 60.41 | 82.29 | 55.49 | 80.12 | 52.79 | 76.26 |
|           | Ours   | 58.32 | 84.90 | 56.38 | 81.77 | 54.87 | 81.07 |
| IEMOCAP   | AE     | 76.15 | 82.09 | 75.07 | 79.84 | 74.20 | 76.91 |
|           | CRA    | 77.05 | 82.13 | 75.95 | 79.95 | 74.22 | 77.03 |
|           | MCTN   | 78.57 | 82.27 | 76.83 | 80.56 | 74.77 | 77.89 |
|           | TransM | 79.57 | 82.64 | 77.21 | 81.13 | 75.87 | 79.01 |
|           | MMIN   | 80.83 | 83.43 | 78.02 | 82.32 | 76.38 | 79.53 |
|           | Ours   | 81.15 | 85.39 | 78.37 | 83.63 | 77.55 | 82.33 |

4.2 Baselines

To evaluate the performance of our approach, the following baselines are chosen for comparison:

- **AE** [2]: An efficient data encoding network trained to copy its input to its output. In our implementation, we employ 5 AEIs with each layer of the size [512, 256, 128, 64].
- **CRA** [33]: A missing modality reconstruction framework that employed the residual connection mechanism to approximate the difference between the input data. In our implementation, we add a residual connection for the input with the same layer setting in AE [2].
- **MCTN** [26]: A method to learn robust joint representations by translating among modalities, claiming that translating from a source modality to a target modality can capture joint information among modalities.
- **TransM** [37]: An end-to-end translation based multimodal fusion method that utilized Transformer to translate among modalities and encoded multimodal features. In our implementation, we concatenate 6 MAE losses between two modalities transformation.
- **MMIN** [49]: A unified multimodal emotion recognition model that adopted the cascade residual auto-encoder and cycle consistency learning to recover missing modalities.

Except for the first to the fifth columns data, we finally obtain 709-dimensional visual representations, where the face, the head, and the eye movements are included.

**Textual Representations:** For each textual utterance, the pre-trained Bert [9] is utilized to extract textual features. Eventually, we adopt the pre-trained uncased BERT-base model (12-layer, 768-hidden, 12-heads) to acquire 768-dimensional word vectors.

**Acoustic Representations:** As an audio analysis toolkit, Librosa [24] shows an excellent ability to extract acoustic features. For CMU-MOSI and IEMOCAP datasets, each audio is mixed to the mono and is re-sampled to 16000 Hz. Besides, each frame is separated by 512 samples, and we choose the zero crossing rate, the mel-frequency cepstral coefficients (MFCC) and the Constant-Q Togram (CQT) features to represent audio segments. Finally, we concatenate three features to yield 33-dimensional acoustic features.

https://github.com/hainow/MCTN
https://github.com/AIM3-RUC/MMIN/tree/master
To explore the effects of different modules in TATE, we evaluate our (MCTN, TransM) achieve better performance, probably due to the robustness of the network. Owing to the forward differential loss and on ACC compared to other baselines, demonstrating the robustness of the network. It can be seen that our proposed model still improves about 0.89% to 3.10% on M-F1 and about 1.31% to 4.81% on ACC compared to other baselines, validating the effectiveness of our model. As can be seen in the table, compared to auto-encoder based methods (AE, CRA), translation-based methods (MCTN, TransM) achieve better performance, probably due to the fact that end-to-end translation among modalities can better fuse the multimodal information. Besides, the comparative experiments suggest that the backward decoder can assist the forward encoder, so as to further improve the overall performance.

For multiple modalities missing cases, we also present related findings in Table 3. In this setting, one or two modalities are randomly discarded. It can be seen that our proposed model still improves about 0.89% to 3.10% on M-F1 and about 1.31% to 4.81% on ACC compared to other baselines, demonstrating the robustness of the network. Owing to the forward differential loss and the assistance of tag, our model can still capture semantic-relevant information. More comparison will be given in Section 4.5.

### 4.4 Ablation Study

To explore the effects of different modules in TATE, we evaluate our model with several settings: 1) using only one modality; 2) using two modalities; 3) removing the tag encoding module; 4) removing the common space projection module; 5) removing the tag recovery loss; 6) removing the forward differential loss; and 7) removing the backward reconstruction loss.

According to Table 4, one interesting finding is that the performance drops sharply when the textual modality is missing, validating that textual information dominates in the multimodal sentiment analysis. A possible explanation for these results is that textual information is the manual transcription. However, similar reductions are not observed when removing the visual modality. We conjecture that the visual information is not well extracted due to the minor changes to the face. Besides, the top half of the table shows that the combination of two modalities provides better performance than single modality, indicating that two modalities can learn complementary features. As for the effects of different modules, the performance of the forward differential module decreases about 4.52% to 6.38% on M-F1 and about 3.69% to 9.09% on ACC compared to the whole model, demonstrating the importance of the forward guidance. Since we employ full modality to pre-train the guidance network, the forward JS divergence loss serves as a good supervision. One striking result to emerge from this table is that the tag encoding module slightly improves the performance as expected. To further validate the effectiveness of the tag encoding module, we conduct several experiments in the following sub-section.

### 4.5 Effects of the Tag Encoding

We incorporate the tag encoding module with two basic models: AE and TransM. The reason why we choose the above two models is that AE and TransM are two different kinds of encoders: AE is the auto-encoder based method, and TransM is the Transformer based method. For the above two models, we add tags after the feature extraction module. Table 5 presents the detailed results on the CMU-MOSI dataset with a 30% missing rate. It can be seen that models with the tag encoding module improves about 2.54% to 3.69% on M-F1 and about 1.04% to 4.08% on ACC compared to basic models, showing the effectiveness of the tag encoding module. Owing to the added tag, the network can be better guided, and can further focus on missing modalities.

**Table 5: Improvements of the Tag Encoding.**

| Model   | Basic +Tag |
|---------|------------|
|         | M-F1 ACC   | M-F1 ACC   |
| AE      | 51.28 72.53 | 53.25 75.21 (3.69% ↑) |
| TransM  | 52.87 72.92 | 54.79 75.21 (3.50% ↑) |
| Ours    | 53.71 79.89 | 55.11 76.02 (2.54% ↑) |

**Table 4: Comparison of all modules in TATE.**

| Modules | M-F1 | ACC | M-F1 | ACC | M-F1 | ACC | M-F1 | ACC | M-F1 | ACC | M-F1 | ACC |
|---------|------|-----|------|-----|------|-----|------|-----|------|-----|------|-----|
| V       | 37.84 | 56.25 | -    | -   | -    | -   | -    | -   | -    | -   | -    | -   |
| A       | 39.82 | 59.90 | -    | -   | -    | -   | -    | -   | -    | -   | -    | -   |
| T       | 55.63 | 76.17 | -    | -   | -    | -   | -    | -   | -    | -   | -    | -   |
| V+A     | 40.71 | 61.26 | 38.93 | 59.10 | 38.07 | 56.94 | 37.42 | 56.18 | 36.98 | 55.43 | 36.66 | 54.17 |
| V+T     | 56.98 | 79.13 | 56.41 | 78.67 | 55.07 | 76.29 | 54.83 | 74.90 | 52.86 | 74.15 | 50.32 | 72.92 |
| A+T     | 57.69 | 80.65 | 57.01 | 79.47 | 55.23 | 77.44 | 55.02 | 75.39 | 53.92 | 74.37 | 51.25 | 73.13 |
| V+T+A   | 58.32 | 84.90 | 58.21 | 84.46 | 55.46 | 81.25 | 55.11 | 80.73 | 54.11 | 80.21 | 51.71 | 74.04 |
| -w/o tag | 57.95 | 80.21 | 57.86 | 80.99 | 54.83 | 79.90 | 53.71 | 79.89 | 52.59 | 76.75 | 49.17 | 72.05 |
| -w/o tag loss | 58.04 | 82.81 | 57.92 | 81.98 | 55.32 | 80.77 | 53.95 | 80.28 | 52.83 | 77.01 | 49.41 | 72.92 |
| -w/o forward loss | 52.39 | 76.21 | 51.83 | 75.52 | 50.16 | 73.39 | 49.87 | 72.15 | 48.26 | 71.12 | 47.29 | 70.35 |
| -w/o backward loss | 53.85 | 77.43 | 52.88 | 77.08 | 51.85 | 74.09 | 51.07 | 74.21 | 49.01 | 71.92 | 48.53 | 71.08 |
| -w/o common space | 54.03 | 79.76 | 53.20 | 77.59 | 52.97 | 75.99 | 51.23 | 75.01 | 49.83 | 72.37 | 49.05 | 71.85 |
4.6 Effects of the complete modality
To see the difference between the complete and incomplete modalities of the test data, we first train the model with incomplete data, and then test the model with both full modality data and different missing rates of incomplete data. All experiments share the same parameters on the IEMOCAP dataset for a fair comparison. As can be seen in Fig. 5, the gaps between two settings on M-F1 and ACC reach the minimum when the missing rate is 0.3. As the number of missing samples in the training data increases, the correlation among modalities becomes harder to capture, resulting in weaker test performance. However, the gap increases when the missing rate is bigger than 0.3. One possible explanation for the above results is that the model cannot learn the joint representation well because there are too many absent samples.

4.7 Multi-classes on IEMOCAP

Table 6: Detailed distributions on IEMOCAP.

| Category | Hap. | Ang. | Sad. | Neu. | Fru. | Exc. | Sur |
|----------|------|------|------|------|------|------|-----|
| 2-classes |      |      |      |      |      |      |     |
| Train    | 477  | 879  | 868  | 1385 | -    | -    | -   |
| Test     | 118  | 224  | 216  | 323  | -    | -    | -   |
| 4-classes |      |      |      |      |      |      |     |
| Train    | 476  | 891  | 873  | 1348 | 1458 | 848  | 87  |
| Test     | 119  | 212  | 211  | 360  | 391  | 193  | 20  |
| 7-classes |      |      |      |      |      |      |     |
| Train    | 1552 | 1552 | 1552 | 1552 | 1552 | 1552 | 1552|
| Test     | 119  | 212  | 211  | 360  | 391  | 193  | 20  |

We also explore the performance of multiple classes on the IEMOCAP dataset. Apart from the two-classes results, we also choose happy, angry, sad and neutral emotions as the 4-classes experiment, and then choose the extra frustration, excited, and surprise emotions as the 7-classes experiment. The detailed distributions and results are presented in Table 6 and Table 7 respectively. It can be seen that both M-F1 value and ACC decrease with the increment of class numbers. By comparing the results with different rates of missing modalities, the gaps among 7-classes are smaller than that among 2-classes and 4-classes. Besides, closer inspection of Table 7 shows that the overall performance drops sharply when the class number is 7, which is caused by the confusion of multiple classes, resulting in the difficulties in convergence of the model.

4.8 Effects of different losses

Table 8: Results of different losses.

| Datasets | Loss | 0   | 0.2 | 0.4 |
|----------|------|-----|-----|-----|
|          | M-F1 | ACC | M-F1 | ACC | M-F1 | ACC |
| CMU-MOSI | Cosine | 56.21 | 82.05 | 52.31 | 76.95 | 50.23 | 74.18 |
|          | MAE   | 57.89 | 84.15 | 54.52 | 78.95 | 52.21 | 75.22 |
|          | JS    | 58.32 | 84.90 | 55.46 | 80.25 | 54.11 | 80.21 |
|          | ours  | 79.86 | 84.07 | 72.21 | 81.69 | 75.48 | 80.06 |
| IEMOCAP  | Cosine | 56.21 | 82.05 | 52.31 | 76.95 | 50.23 | 74.18 |
|          | MAE   | 57.89 | 84.15 | 54.52 | 78.95 | 52.21 | 75.22 |
|          | JS    | 58.32 | 84.90 | 55.46 | 80.25 | 54.11 | 80.21 |
|          | ours  | 80.21 | 84.21 | 77.67 | 82.53 | 75.89 | 80.21 |

To investigate the effects of different losses, we replace different loss function to see the performance. In detail, the cosine similarity loss, the MAE loss, and the JS divergence loss are chosen for comparison. We evaluate our model with 4 settings: 1) using the cosine similarity loss for $L_{\text{forward}}, L_{\text{backward}}$ and $L_{\text{tag}}$; 2) using the MAE loss for $L_{\text{forward}}, L_{\text{backward}}$ and $L_{\text{tag}}$; 3) using the JS divergence loss for $L_{\text{forward}}, L_{\text{backward}}$ and $L_{\text{tag}}$; and 4) using the JS divergence loss for $L_{\text{forward}}$ and $L_{\text{backward}}$, and using the MAE loss for $L_{\text{tag}}$.

As can be seen in the Table 8, our method achieves the best performance compared to other three loss settings on two datasets, showing the superiority of our model. Relatively, the results of applying JS divergence achieve secondary performance. Since the tag is composed of 4 digits ("0" or "1"), the MAE loss is more straightforward than JS divergence loss. Further analysis of the table suggests that the combination of the JS divergence loss and the MAE loss is beneficial in improving the overall performance.

4.9 Visualization
To better understand the learning ability of our model, we adopt the T-SNE toolkit [20] to visualize the joint representations under different rates of missing modalities. Specifically, we visualize about 1500 vectors learned by the Transformer encoder on the CMU-MOSI dataset, where the red, the blue, and the green color denote negative, neutral and positive respectively.

As shown in Figs. 6(a)-(e), the overall joint representations obtain the similar distribution as the full modality condition. The majority of vectors are generally divided into three categories, where neutral samples is harder to classify because of their uncertain semantic. Besides, with the increment of missing ratio, the distributions become more discrete, especially when the missing ratio is bigger than 0.3. Apart from that, as can be seen in the top right-hand corner of Figs. 6(b)-(e), the larger ratio of missing modalities, the wider
outliers. The reason is that the model cannot converge with too many absent samples. While in Fig. 6(f), the decision boundary is closer to the outliers when there are nearly half of missing samples. We suspect that absent samples dominate when training the model, resulting in a quite distinct distribution.

5 CONCLUSION

In this paper, we propose a Tag-Assisted Transformer Encoder (TATE) network to handle the problem of missing partial modalities. Owing to the tag encoding technique, the proposed model can cover all uncertain missing cases, and the designed tag recovery loss can in turn supervise joint representation learning. Besides, more general aligned vectors are obtained by the common space module, and then are fed into the Transformer encoder for further process. At last, the final objective loss further directs the learning of missing modalities. All experimental results are conducted on CMU-MOSI and IEMOCAP datasets, showing the effectiveness of the proposed method.

In the future, this study may be further improved in the following ways: 1) for the common space projection module, we will try more fusion methods (e.g., add weights) to concatenate common feature vectors; and 2) for the Transformer encoder-decoder module, we employ the original sub-layer in Transformer as the basic semantic encoder. We attempt to adopt different structures of Transformer (e.g., Transformer-XL [8], Reformer [19], etc.) to observe the performance.

ACKNOWLEDGMENTS

This work was supported by Macau Science and Technology Development Fund under SKL-IOTSC-2021-2023, 0072/2020/AMJ, 0015/-2019/AKP, 060/2019/A1, and 077/2018/A2, by Research Committee at University of Macau under MYRG2018-00029-FST and MYRG2019-00023-FST, and by Natural Science Foundation of China under 61971476.

REFERENCES

[1] Hassan Akbari, Linagzhe Yuan, Rui Qian, Wei-Hong Chuang, Shih-Fu Chang, Yin Cui, and Boqing Gong. 2021. Vatt: Transformers for multimodal self-supervised learning from raw video, audio and text. In Proceedings of the 35th Annual Conference on Neural Information Processing Systems.
[2] Pierre Baldi. 2012. Autoencoders, unsupervised learning, and deep architectures. In Proceedings of ICML Workshop on Unsupervised and Transfer Learning. 37–49.
[3] Tadas Baltrusaitis, Amir Zadeh, Yao Chong Lim, and Louis-Philippe Morency. 2018. Openface 2.0: Facial behavior analysis toolkit. In 2018 13th IEEE International Conference on Automatic Face & Gesture Recognition. IEEE, 59–66.
[4] Carlos Busso, Murtaza Bafat, Chi-Chun Lee, Abe Kazemzadeh, Emily Mower, Samuel Kim, Jeannette N Chang, Sungbok Lee, and Shrikanth S Narayanan. 2008. IEMOCAP: Interactive emotional dyadic motion capture database. Language Resources and Evaluation 42, 4 (2008), 335–359.
[5] 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. 1158–1166.
[6] Mingzhe Chen, Sen Wang, Paul Fu Liang, Tadas Baltrusaitis, Amir Zadeh, and Louis-Philippe Morency. 2017. Multimodal sentiment analysis with word-level fusion and reinforcement learning. In Proceedings of the 2017 ACM International Conference on MULTIMODAL Interaction. 163–171.
[7] Xiangxiang Chu, Zhi Tian, Yuqing Wang, Bo Zhang, Haibing Ren, Xiaolin Wei, Huaxia Xia, and Chunhua Shen. 2021. Twins: Revisiting the Design of Spatial
Attention in Vision Transformers. In Advances in Neural Information Processing Systems, Vol. 34. Curran Associates, Inc., 9355–9366.

[8] Zháňg Dái, Zhián Yang, Yímáng Yang, Jíáíe Carbénnel, Quoé V Le, and Ruzán Sąakkłdtudnó. 2018. Transformer-xl: Attesting language models beyond a fixed-length context. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. 2978–2988.

[9] Jacob Devlin, Ming-Ni 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. 4171–4186.

[10] Changde Du, Changying Du, Hao Wang, Junpeng Li, Wei-Long Zheng, Bao-Liang Lu, and Huaiqiang He. 2018. Semi-supervised deep generative modeling of incomplete multi-modality emotional data. In Proceedings of the 26th ACM International Conference on Multimedia. 108–116.

[11] Binzong Geng, Min Yang, Fajie Yuan, Shupeng Wang, Xiang Ao, and Ruifeng Xu. 2021. Iterative network pruning with uncertainty regularization for lifelong sentiment classification. In Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval. 1229–1238.

[12] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. Advances in Neural Information Processing Systems 27 (2014).

[13] Dan Guo, Wengang Zhou, Houqiang Li, and Meng Wang. 2017. Online early-late fusion based on adaptive HMM for sign language recognition. ACM Transactions on Multimedia Computing, Communications, and Applications 14, 1 (2017), 1–18.

[14] Devamanyu Hazarika, Roger Zimmermann, and Soujanya Poria. 2020. Misa: Modality-invariant-and-specific representations for multimodal sentiment analysis. In Proceedings of the 28th ACM International Conference on Multimedia. 1122–1131.

[15] Kaining He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 770–778.

[16] Eun-Sol Kim, Woon Young Kang, Kyoung-Woon On, Ju-Yung Heo, and Byoung-Tak Zhang. 2020. Hypergraph attention networks for multimodal learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 14591–14599.

[17] Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In Proceedings of the 3rd International Conference on Learning Representations. 1–15.

[18] Diederik P Kingma and Max Welling. 2014. Auto-encoding variational bayes. In Proceedings of the International Conference on Machine Learning. 1–12.

[19] Laurens van der Maat en and Geoffrey Hinton. 2008. Visualizing data using t-SNE. Journal of Machine Learning Research 9, Nov (2009), 2579–2605.

[20] Siyé Mai, Songlong Jiang, and Haifeng Hu. 2021. Analyzing multimodal sentiment via acoustic-and-visual-LSTM with channel-aware temporal convolution network. IEEE/ACM Transactions on Audio, Speech, and Language Processing 29 (2021), 1424–1437.

[21] 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.

[22] Rahul Manzumder, Trevor Hastie, and Robert Tibshirani. 2010. Spectral regularization algorithms for learning large incomplete matrices. The Journal of Machine Learning Research 11 (2010), 2287–2322.

[23] Brian McFee, Colin Raffel, Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Dierks, and Soujanya Poria. 2018. Multimodal sentiment analysis using hierarchical fusion with context modeling. Knowledge-based Systems 161 (2018), 124–133.

[24] Rabih Manzumder, Trevor Hastie, and Robert Tibshirani. 2010. Spectral regularization algorithms for learning large incomplete matrices. The Journal of Machine Learning Research 11 (2010), 2287–2322.

[25] Brian McFee, Colin Raffel, Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Dierks, and Soujanya Poria. 2018. Multimodal sentiment analysis using hierarchical fusion with context modeling. Knowledge-based Systems 161 (2018), 124–133.

[26] Brian McFee, Colin Raffel, Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Dierks, and Soujanya Poria. 2018. Multimodal sentiment analysis using hierarchical fusion with context modeling. Knowledge-based Systems 161 (2018), 124–133.