A Wander Through the Multimodal Landscape: Efficient Transfer Learning via Low-rank Sequence Multimodal Adapter

Zirun Guo, Xize Cheng, Yangyang Wu, Tao Jin*

Zhejiang University
zrguo.cs@gmail.com

Abstract

Efficient transfer learning methods such as adapter-based methods have shown great success in unimodal models and vision-language models. However, existing methods have two main challenges in fine-tuning multimodal models. Firstly, they are designed for vision-language tasks and fail to extend to situations where there are more than two modalities. Secondly, they exhibit limited exploitation of interactions between modalities and lack efficiency. To address these issues, in this paper, we propose the lo**W**-r**a**nk sequ**e**nce multimo**d**al adapt**er** (**Wander**). We first use the outer product to fuse the information from different modalities in an element-wise way effectively. For efficiency, we use CP decomposition to factorize tensors into rank-one components and achieve substantial parameter reduction. Furthermore, we implement a token-level low-rank decomposition to extract more fine-grained features and sequence relationships between modalities. With these designs, Wander enables token-level interactions between sequences of different modalities in a parameter-efficient way. We conduct extensive experiments on datasets with different numbers of modalities, where Wander outperforms state-of-the-art efficient transfer learning methods consistently. The results fully demonstrate the effectiveness, efficiency and universality of Wander.

1 Introduction

In recent years, multimodal models, such as BLIP-2 (Li et al. 2023a), LLaVA (Liu et al. 2024), have experienced rapid development and have shown excellent performance in various downstream tasks. However, the increasing number of parameters in these multimodal models, coupled with the diversification of downstream tasks, has resulted in a significant consumption of computational resources and time for fine-tuning these models. Consequently, efficient transfer learning strategies (Hu et al. 2021; Houlsby et al. 2019; Vu et al. 2021; Liu et al. 2023a; Dettmers et al. 2024) have become a focal point of current research.

For multimodal models, there are two kinds of popular efficient transfer learning strategies, including adapter-based methods (Zhang et al. 2021; Sung, Cho, and Bansal 2022; Gao et al. 2024; Lu et al. 2024) and prompt learning methods (Zang et al. 2022; Xing et al. 2023; Khattak et al. 2023; Guo, Jin, and Zhao 2024; Yan et al. 2024). Adapter-based methods add extra modules and only train these added parameters while freezing the pre-trained model. For example, Lu et al. (2024) propose a unified and knowledge-sharing design to interact between the vision and language modalities. Prompt learning methods insert trainable prompts to the input or attention matrices. For instance, Khattak et al. (2023) propose a coupling function to explicitly condition vision prompts on their language counterparts, which acts as a bridge between the two modalities.

Nevertheless, there are two main challenges currently associated with efficient transfer learning for multimodal models. Firstly, existing multimodal transfer learning techniques (Sung, Cho, and Bansal 2022; Lu et al. 2024) are primarily limited to fine-tuning visual-language models, focusing only on the interaction between these two modalities and failing to extend to multimodal models with additional modalities. Secondly, the existing transfer learning strategies for multimodal models exhibit limited exploitation of interactions between modalities and lack efficiency when applied to models with multiple modalities. Specifically, existing multimodal efficient transfer learning strategies focus on how to fuse vector representations from different modalities rather than sequences of vector representations from different modalities, overlooking the interactions of time dimensions of various modalities.

Based on the above observations, in this paper, we propose the lo**W**-r**a**nk sequ**e**nce multimo**d**al adapt**er** (**Wander**) for efficient multimodal transfer learning. Wander enables fine-grained token-level interactions between sequences of different modalities in a parameter-efficient way. Specifically, motivated by the outer product fusion and low-rank decomposition, we fuse the information from different modalities in an element-wise and token-level way. Furthermore, we utilize CP decomposition to factorize tensors into low-rank components and achieve a substantial reduction in the number of parameters. We conduct extensive experiments on four datasets with different numbers of modalities. The performances demonstrate the effectiveness and efficiency of Wander. In summary, our contributions are as follows:

- We propose the low-rank sequence multimodal adapter that can be applied to situations with any number of modalities in a parameter-efficient way.
- Wander enables fine-grained token-level interactions be-

*Corresponding author

tween sequences of different modalities.

- Wander outperforms other efficient transfer learning methods consistently with fewer parameters on all the datasets.

2 Related Work

Efficient Transfer Learning. Efficient transfer learning offers a way to fine-tune the model with much fewer parameters and lower computational resources than full finetuning while achieving comparable performance. Efficient transfer learning can be mainly divided into two groups: additive fine-tuning methods and LoRA-based methods. Additive finetuning methods, which can be further divided into adapter methods (Houlsby et al. 2019; Sung, Cho, and Bansal 2022; Lu et al. 2024) and prompt methods (Liu et al. 2023a; Li and Liang 2021; Vu et al. 2021; Wang et al. 2024; Guo, Jin, and Zhao 2024; Yan et al. 2024), add extra modules or parameters to the model and only train these added parameters while freezing the large pre-trained model. LoRA-based methods (Hu et al. 2021; Dettmers et al. 2024; Li et al. 2024; Chen et al. 2024), which are also referred to as reparameterization methods, construct low-rank weight matrices, add them to existing weights and only train these low-rank matrices. For multimodal learning (Guo et al. 2024), adapter-based methods and prompt learning are popular. Khattak et al. (2023) propose a coupling function to explicitly condition vision prompts on their language counterparts. Lu et al. (2024) propose a knowledge-sharing adapter design which enables efficient adaptation with cross-modal representations. Wang et al. (2024) propose a prompt framework which utilizes mode approximation to implement multimodal efficient transfer learning. However, these methods are designed for vision-language models and fail to extend to situations where there are more than two modalities. Besides, these methods focus on sequence coarse-grained features, exhibiting limited exploitation of interactions between modalities. In contrast, our method can be applied to tasks with any number of modalities with token-level and fine-grained interactions between modalities.

Multimodal Fusion. Multimodal fusion can be divided into early fusion, late fusion and intermediate fusion. Early fusion methods (Liu et al. 2023b; Liang et al. 2022) integrate multimodal features through concatenation or a simple function such as averaging before inputting into the models. Late fusion methods (Tsai et al. 2019) use data from different modalities independently followed by fusion at a decision-making stage. Intermediate fusion methods (Zadeh et al. 2017; Liu et al. 2018; Pérez-Rúa et al. 2019; Joze et al. 2020) allow data fusion at different stages of model training. For example, Zadeh et al. (2017) use the outer product to model element-wise features between modalities at different stages of the model. Compared with intermediate fusion strategies, early fusion and late fusion methods can not model inter-modality information effectively and are not as flexible as intermediate methods. However, intermediate methods have more parameters and existing intermediate methods focus on sequence-level features. In contrast, our method can model token-level features between sequences from different modalities in a parameter-efficient way by using low-rank factors.

3 Methodology

In this section, we will introduce our lo**W**-r**a**nk seque**n**ce multimo**d**al adapt**er** (**Wander**) for efficient multimodal transfer learning. We will first introduce preliminaries in Section 3.1. Then, we will make a simple analysis of the outer product fusion in Section 3.2 before introducing our Wander architecture in Section 3.3.

3.1 Preliminaries

Adapter Tuning. Tuning with adapter (Houlsby et al. 2019) modules involves adding a small number of new parameters to a model. An adapter module consists of a feedforward down-projection layer, a nonlinearity and an up-projection layer. Besides, a skip connection is added. We can represent the adapter module as:

$$\texttt{Adapter}(x) = x + \texttt{Up}(\texttt{Nonlinear}(\texttt{Down}(x))) \quad (1)$$

where $\texttt{Up}(\cdot)$, $\texttt{Nonlinear}(\cdot)$ and $\texttt{Down}(\cdot)$ represent the up-projection layer, nonlinear function and down-projection layer, respectively.

CP decomposition. The CANDECOMP/PARAFAC or canonical polyadic (CP) decomposition factorizes a tensor into a sum of outer products of vectors. Given an N-dimensional tensor $\mathcal{X} \in \mathbb{R}^{d_1 \times d_2 \times \cdots \times d_N}$, it can be represented as a combination of tensors:

$$\mathcal{X} = \sum_{r=1}^{R} \bigotimes_{n=1}^{N} a_n^r \quad (2)$$

where R is the rank and $a_n^r \in \mathbb{R}^{d_n}$. $\bigotimes_{n=1}^{N}$ is the tensor outer product operation over a set of vectors indexed by n.

3.2 Outer Product Multimodal Fusion

Multimodal fusion allows interaction between different modalities. Early fusion and late fusion are two common strategies. However, these methods simply integrate the inputs or outputs from different modalities using weighted averaging or several MLP layers, which can not model inter-modality interactions effectively (Liu et al. 2018). Therefore, more intermediate approaches (Zadeh et al. 2017; Liu et al. 2018; Pérez-Rúa et al. 2019; Joze et al. 2020) are proposed. One notable category of methods computes the outer product between unimodal representations (Zadeh et al. 2017) which has shown great success. Given M modalities, we denote them as $m_1, m_2, \cdots, m_M$ and the representation of each modality as $h_1, h_2, \cdots, h_M$. The unimodal representation $h_i \in \mathbb{R}^{d_i}$ where $i = 1, 2, \cdots, M$ and d_i is the dimension. Then outer product method integrates these representations into a multimodal representation H as follows:

$$H = \bigotimes_{m=1}^{M} h_m \quad (3)$$

where $\bigotimes_{m=1}^{M}$ is the tensor outer product operation over a set of vectors indexed by m. After outer product operation, the multimodal representation $H \in \mathbb{R}^{d_1 \times d_2 \times \cdots \times d_M}$ is projected into a vector representation $\tilde{H}$ using a linear layer:

$$\tilde{H} = \mathbf{W} \cdot H + b = \mathbf{W} \cdot \bigotimes_{m=1}^{M} h_m + b, \quad \tilde{H} \in \mathbb{R}^{d_h} \quad (4)$$

where $\mathbf{W}$ is the weight matrix and b is the bias of the linear layer. d_h represents the dimension of the projected multimodal vector. The weight matrix $\mathbf{W} \in \mathbb{R}^{d_1 \times d_2 \times \cdots \times d_M \times d_h}$.

Compared with other fusion methods such as late fusion methods, outer product operation enables element-wise interactions between different modalities, thus achieving better results. However, as Equation 3 and 4 shows, outer product operation needs to explicitly calculate the high-dimensional tensor H and needs a high-dimensional matrix $\mathbf{W}$ to project the multimodal representation H into the vector $\tilde{H}$, where the number of parameters will increase exponentially with the number of modalities which needs lots of computational resources.

3.3 Low-rank Sequence Multimodal Adapter

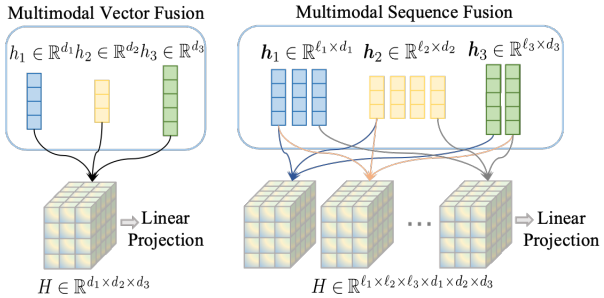
Motivation As aforementioned, existing efficient transfer learning methods for multimodal tasks have two main limitations. On the one hand, these methods (Lu et al. 2024; Gao et al. 2024; Zhang et al. 2021; Sung, Cho, and Bansal 2022) are designed for vision-language models and can not be extended to situations where there are more than two modalities. On the other, they focus on integrating vectors from different modalities rather than sequences of vectors which are fine-grained features (Figure 1). Motivated by outer product fusion introduced in Section 3.2 and the low-rank multimodal fusion method (Liu et al. 2018), we focus on addressing the two limitations and propose the Wander.

As discussed in Section 3.2, outer product fusion can model inter-modality interactions effectively because it enables element-wise calculation between modality vectors. However, it can not be applied to efficient transfer learning for multimodal tasks due to the need for heavy computational resources. Besides, it only models one vector representation $h \in \mathbb{R}^d$ where d is the dimension of the representation, not sequences of vector representations from different modalities. Therefore, we need to make outer product fusion more efficient and enable it to deal with sequences of representations. To explain the low-rank sequence multimodal adapter more clearly, we first introduce low-rank single vector fusion before the sequence vector fusion.

Vector fusion We first start by introducing the low-rank fusion of single vector representations from different modalities. According to Equation 4, we denote the linear projection matrix as $\mathbf{W}_h \in \mathbb{R}^{d_1 \times d_2 \times \cdots \times d_M \times d_h}$ which is an (M+1)-order tensor. We partition $\mathbf{W}_h$ into $\mathbf{W}_h^k \in \mathbb{R}^{d_1 \times d_2 \times \cdots \times d_M}, k = 1, 2, \cdots, d_h$. Then according to CP decomposition which factorizes a tensor into a sum of component rank-one tensors, we can transform $\mathbf{W}_h^k$ into a low-rank form:

$$\mathbf{W}_h^k = \sum_{r=1}^{R} \bigotimes_{m=1}^{M} w_{h,m,k}^r, w_{h,m,k}^r \in \mathbb{R}^{d_m} \tag{5}$$

where R is a positive integer denoting the rank of the tensor. In other words, we can reconstruct $\mathbf{W}_k$ with these rank-one tensors $\{\{w_{h,m,k}^r\}_{m=1}^M\}_{r=1}^R$. We regroup these tensors into M modality rank-one tensors. For a specific modality m, we can denote its decomposition factors as $\mathbf{w}_m = [\mathbf{w}_{h,m}^1, \mathbf{w}_{h,m}^2, \cdots, \mathbf{w}_{h,m}^R]$ where $\mathbf{w}_{h,m}^r = [w_{h,m,1}^r, w_{h,m,2}^r, \cdots, w_{h,m,d_h}^r]$ and $\mathbf{w}_{h,m}^r \in \mathbb{R}^{d_h \times d_m}$. Therefore, we can write the linear matrix $\mathbf{W}$ as follows:



(a) Vector Fusion (b) Sequence Fusion

Figure 1: The difference between vector fusion and sequence fusion in their original outer product form. Sequence fusion enables token-level interactions between modalities. We take three modalities as an example.

$$\mathbf{W}_h = \sum_{r=1}^{R} \bigotimes_{m=1}^{M} \mathbf{w}_{h,m}^r \tag{6}$$

Then, based on the properties of outer products, summation, and element-wise multiplication, Equation 4 can be rewritten as:

$$\begin{aligned} \tilde{H} &= \sum_{r=1}^{R} \left(\bigotimes_{m=1}^{M} \mathbf{w}_{h,m}^r \cdot \bigotimes_{m=1}^{M} h_m \right) + b_h \\ &= \bigwedge_{m=1}^{M} \left[\sum_{r=1}^{R} \mathbf{w}_{h,m}^r \cdot h_m \right] + b_h \end{aligned} \tag{7}$$

where $\Lambda_{m=1}^M$ denotes the element-wise multiplication over a sequence of vectors indexed by m. Compared with Equation 4, Equation 7 does not compute the outer product explicitly which reduces the computational complexity. Besides, the parameter matrix $\mathbf{W}_h \in \mathbb{R}^{d_1 \times d_2 \times \cdots \times d_M \times d_h}$ is now transformed into R 2-order matrix $\mathbf{w}_{h,m}^r \in \mathbb{R}^{d_h \times d_m}$, indicating a substantial reduction in the number of parameters.

Sequence fusion Vector fusion overlooks the fact that the Transformer outputs sequences of vectors which prevents it from modeling token-level and fine-grained features. Therefore, we propose sequence fusion to model these features in Transformers. The difference between vector fusion and sequence fusion is shown in Figure 1. Given multimodal representations $\boldsymbol{h}_1, \boldsymbol{h}_2, \cdots, \boldsymbol{h}_M$ where M denotes the number of modalities and each representation $\boldsymbol{h}_m \in \mathbb{R}^{\ell_m \times d_m}, m = 1, 2, \cdots, M$ consists of a sequence of vectors outputted by the Transformer layer. d_m is the dimension of the vector and ℓ_m is the sequence length. Concretely, $\boldsymbol{h}_m = [h_m^1, h_m^2, \cdots, h_m^{\ell_m}]$ where $h_m^i \in \mathbb{R}^{d_m}, i = 1, 2, \cdots, \ell_m$. Given a vector representation h_m^i, we want it to interact with all the representations from all the other modalities. Specifically, we use the following equation to represent the process:

$$H_{i_1, i_2, \cdots, i_M} = \mathrm{VF}(h_1^{i_1}, h_2^{i_2}, \cdots, h_M^{i_M}), \quad H_{i_1, i_2, \cdots, i_M} \in \mathbb{R}^{d_h} \tag{8}$$

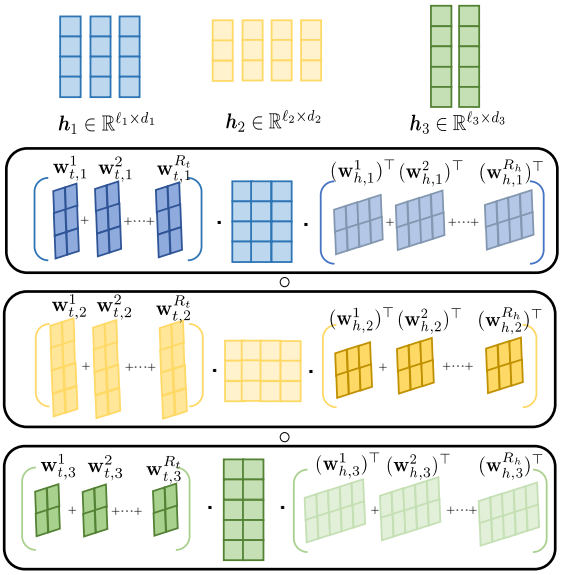


Figure 2: The illustration of low-rank sequence fusion. We use three modalities as an example. $\circ$ denotes element-wise multiplication.

where VF denotes the vector fusion process in Section 3.3, d_h is the dimension and $h_m^{i_m}$ denotes the i_m-th representation in $\boldsymbol{h}_m, m = 1, 2, \cdots, M, i_m = 1, 2, \cdots, \ell_m$. $H_{i_1,i_2,\cdots,i_M}$ denotes the integrated representation of the corresponding representations from different modalities. From Equation 8, we can observe that we will do vector fusion $\prod_{m=1}^{M} \ell_m$ times to get the final representation $H_t \in \mathbb{R}^{\ell_1 \times \ell_2 \times \cdots \times \ell_M \times d_h}$. We can represent H_t as follows:

$$H_t = H \cdot \mathbf{W}_h = \bigotimes_{m=1}^{M} \boldsymbol{h}_m \cdot \mathbf{W}_h \tag{9}$$

where $\mathbf{W}_h \in \mathbb{R}^{d_1 \times d_2 \times \cdots \times d_M \times d_h}$. In the following context, we will omit the bias term for simplicity. Then following Equation 4, we use a linear layer to project H_t into a sequence representation $\tilde{H}_t$:

$$\tilde{H}_t = \mathbf{W}_t \cdot H_t = \mathbf{W}_t \cdot H \cdot \mathbf{W}_h, \quad \tilde{H}_t \in \mathbb{R}^{d_t \times d_h} \tag{10}$$

where the matrix $\mathbf{W}_t \in \mathbb{R}^{d_t \times \ell_1 \times \ell_2 \times \cdots \times \ell_M}$ and d_t is the integrated sequence length. However, this poses the same problem as Equation 4, where the number of parameters of $\mathbf{W}_t$ is large and we need to explicitly calculate the high dimensional tensor H. To reduce the number of parameters, we also use CP decomposition to transform the matrix $\mathbf{W}_t$ into a series of rank-one tensors. Firstly, we partition $\mathbf{W}_t$ into $\mathbf{W}_t^k \in \mathbb{R}^{\ell_1 \times \ell_2 \times \cdots \times \ell_M}, k = 1, 2, \cdots, d_t$. According to CP decomposition, we have:

$$\mathbf{W}_t^k = \sum_{r=1}^{R} \bigotimes_{m=1}^{M} w_{t,m,k}^r, w_{t,m,k}^r \in \mathbb{R}^{\ell_m} \tag{11}$$

Similarly, we regroup these tensors $\mathbf{W}_t^k$ into M modality rank-one tensors. We denote modality m decomposition factors as $\mathbf{w}_{t,m} = [\mathbf{w}_{t,m}^1, \mathbf{w}_{t,m}^2, \cdots, \mathbf{w}_{t,m}^R]$ where $\mathbf{w}_{t,m}^r = [w_{t,m,1}^r, w_{t,m,2}^r, \cdots, w_{t,m,d_t}^r]$ and $\mathbf{w}_{t,m}^r \in \mathbb{R}^{d_t \times \ell_m}$. Then, $\mathbf{W}_t$ can be rewritten as:

$$\mathbf{W}_t = \sum_{r=1}^{R} \bigotimes_{m=1}^{M} \mathbf{w}_{t,m}^r \tag{12}$$

Then, using Equation 6 and 12, we can rewrite Equation 10 as follows:

$$\begin{aligned}
\tilde{H}_t &= \sum_{r_t=1}^{R_t} \bigotimes_{m=1}^{M} \mathbf{w}_{t,m}^{r_t} \left[\sum_{r_h=1}^{R_h} \left(\bigotimes_{m=1}^{M} \mathbf{w}_{h,m}^{r_h} \cdot \bigotimes_{m=1}^{M} \mathbf{h}_m^\top \right) \right]^\top \\
&= \sum_{r_t=1}^{R_t} \sum_{r_h=1}^{R_h} \left(\bigotimes_{m=1}^{M} \mathbf{w}_{t,m}^{r_t} \cdot \bigotimes_{m=1}^{M} \mathbf{h}_m \cdot \bigotimes_{m=1}^{M} (\mathbf{w}_{h,m}^{r_h})^\top \right) \\
&= \bigwedge_{m=1}^{M} \left[\sum_{r_t=1}^{R_t} \sum_{r_h=1}^{R_h} \mathbf{w}_{t,m}^{r_t} \cdot \mathbf{h}_m \cdot (\mathbf{w}_{h,m}^{r_h})^\top \right]
\end{aligned} \tag{13}$$

where $\mathbf{w}_{h,m}^{r_h} \in \mathbb{R}^{d_h \times d_m}, \mathbf{h}_m \in \mathbb{R}^{\ell_m \times d_m}$ and $\mathbf{w}_{t,m}^{r_t} \in \mathbb{R}^{d_t \times \ell_m}$, $\mathbf{w}^\top$ represents the transpose of $\mathbf{w}$, and R_t, R_h denotes the rank in Equation 6 and 12, respectively. From Equation 13, we can observe that it is no longer necessary to calculate $\bigotimes_{m=1}^{M} \boldsymbol{h}_m$ explicitly and the original high-dimensional matrices $\mathbf{W}_h$ and $\mathbf{W}_t$ are transformed into several low-rank 2D matrices. Figure 2 presents the illustration of low-rank sequence fusion.

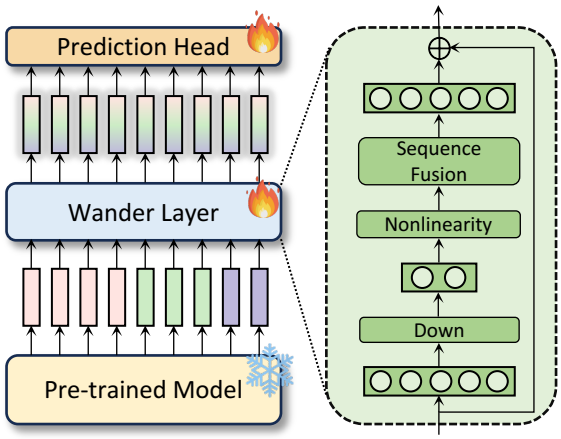


Figure 3: The overall architecture of Wander and its integration with the pre-trained model. **Left:** We add Wander to the pre-trained model for fine-tuning. **Right:** Wander consists of a linear down-projection layer (Down), a nonlinear function, sequence fusion and a skip connection.

Wander Architecture Through Equation 13, we can integrate features from different modalities in a sequence token-level and parameter-efficient way, which can be extended to any number of modalities scenarios. Figure 3 presents the overall architecture of Wander. Sequence Fusion is the main component of Wander, which enables sequence token-level interactions between modalities in a parameter-efficient way. Specifically, we denote the output of the pre-trained Transformer as $\boldsymbol{h}_1, \boldsymbol{h}_2, \cdots, \boldsymbol{h}_M$ where M denotes the number

of modalities and $\boldsymbol{h}_m \in \mathbb{R}^{\ell_m \times d_m}, m = 1, 2, \cdots, M$. For simplicity, we denote $\boldsymbol{h} = [\boldsymbol{h}_1, \boldsymbol{h}_2, \cdots, \boldsymbol{h}_M]$. Then, we can represent the process of Wander as:

$$\texttt{Wander}(\boldsymbol{h}) = \boldsymbol{h} + \texttt{SF}(\texttt{Nonlinear}(\texttt{Down}(\boldsymbol{h}))) \quad (14)$$

where $\texttt{Down}(\cdot)$ is the linear layer, $\texttt{SF}(\cdot)$ is the sequence fusion and $\texttt{Nonlinear}(\cdot)$ is the nonlinear function such as ReLU. Different from the original adapter module (Houlsby et al. 2019), Wander does not have an up-projection layer because the sequence fusion has a projection layer internally. Therefore, we discard the original explicit up-projection layer, which makes Wander more parameter-efficient. Besides, the down-projection layer and sequence fusion are both linear transformations and thus we add a nonlinear function. In this paper, we use the same nonlinear function as the pre-trained model. In the fine-tuning stage, we freeze the pre-trained model and only train the Wander module and the prediction head which is usually a linear layer.

4 Experiments

4.1 Experimental Settings

Datasets and Evaluation Metrics. We evaluate Wander on four different downstream tasks with different numbers of modalities, including UPMC-Food 101 (Wang et al. 2015) (2 modalities), CMU-MOSI (Zadeh et al. 2016) (3 modalities), IEMOCAP (Busso et al. 2008) (3 modalities), MSRVTT (Xu et al. 2016) (7 modalities). Particularly, to evaluate Wander in situations with more modalities, we use extracted features which consist of seven modalities (Gabeur et al. 2020) for MSRVTT.

UPMC-Food 101 is a food classification dataset, which contains about 100,000 recipes for a total of 101 food categories. Each item in the dataset is represented by one image plus textual information.

CMU-MOSI is a popular dataset for multimodal (audio, text and video) sentiment analysis. These videos are carefully selected from YouTube and divided into 2,199 segments. Each segment is manually annotated with a sentiment score ranging from strongly negative to strongly positive (-3 to +3).

IEMOCAP is a multimodal (audio, text and video) emotion recognition dataset, which contains recorded videos from ten actors in five dyadic conversation sessions. Actors engaged in five different scenarios, performing selected emotional scripts and eliciting specific types of emotions (happiness, anger, sadness, frustration and neutral state).

MSRVTT is characterized by unique properties including large-scale clip-sentence pairs, comprehensive video categories, diverse video content and descriptions, as well as multimodal audio and video streams. MSRVTT consists of 10,000 video clips from 20 categories, and each video clip is annotated with 20 English sentences. To validate Wander in situations with more modalities, we use extracted features which have seven modalities. Specifically, following previous work (Gabeur et al. 2020), we extract motion, audio, scene, ocr, face, speech and appearance features.

For UPMC-Food 101 and IEMOCAP, we use binary accuracy (ACC) and F1 score (F1) to evaluate the performance. For CMU-MOSI, we use binary accuracy (ACC-2), F1 score (F1), 7-class accuracy (ACC-7), mean absolute error (MAE, lower is better) and Pearson correlation (Corr, higher is better) to evaluate the performance. For MSRVTT, we use recall at rank N (R@N, higher is better), median rank (MdR, lower is better) and mean rank (MnR, lower is better) to evaluate the performance.

Method	#Tunable	ACC	F1
Full fine-tuning	220M	**91.8**	**91.9**
LoRA (r=64) (Hu et al. 2021)	26.4M	88.7	88.7
Adapter (d=64) (Houlsby et al. 2019)	24.9M	88.4	84.3
Adapter (d=128) (Houlsby et al. 2019)	50.1M	89.2	89.3
P-tuning(Liu et al. 2023a)	6.2M	83.1	83.0
MaPLe (Khattak et al. 2023)	6.5M	84.6	84.7
PMF (Li et al. 2023b)	5.2M	89.1	89.1
UniAdapter (d=128) (Lu et al. 2024)	35.9M	90.8	90.8
Aurora (d=128) (Wang et al. 2024)	**2.6M**	90.2	90.1
Wander (d=64)	3.1M	91.1	91.1
Wander (d=128)	4.9M	**91.8**	91.8

Table 1: Comparison of performance on the UPMC-Food 101 dataset. #Tunable denotes the number of trainable parameters, including the added adapter and the task prediction head. "d=" denotes the down projection dimension of the adapter.

Method	#Tunable	ACC	F1
Full fine-tuning	80M	**74.8**	74.3
LoRA (r=16) (Hu et al. 2021)	4.0M	71.5	71.1
Adapter (d=16) (Houlsby et al. 2019)	4.0M	70.8	71.7
Adapter (d=64) (Houlsby et al. 2019)	15.8M	72.0	71.7
P-tuning(Liu et al. 2023a)	5.0M	72.5	72.4
MaPLe (Khattak et al. 2023)	5.2M	73.2	73.1
PMF (Li et al. 2023b)	4.8M	72.9	72.6
Wander (d=16)	**0.3M**	74.2	73.8
Wander (d=64)	1.0M	74.7	**74.4**

Table 2: Comparison of performance on IEMOCAP.

Implementation Details. For UPMC-Food 101, we use the pre-trained bert-base model (Devlin et al. 2018) and pre-trained ViT (Dosovitskiy et al. 2020) as the backbone. For CMU-MOSI and IEMOCAP, we use stacked Transformer layers as the backbone. For MSRVTT, we use the video retrieval model MMT (Gabeur et al. 2020) as the backbone. For CMU-MOSI, IEMOCAP and MMT, we pre-train the model on the HowTo100M dataset (Miech et al. 2019). In the training process, we add Wander and the task prediction head to the backbone, keep the pre-trained backbone frozen and only train the Wander and the task prediction head. For all the datasets, we set R_h and R_t to 8 by default. For UPMC-Food 101, we set the batch size to 128 and use the AdamW optimizer with a StepLR scheduler where the initial learning rate is 2e-3, step is 30 and the decay rate $\gamma = 0.1$. For CMU-MOSI and IEMOCAP, we set the batch size to 24 and use the Adam optimizer with a StepLR scheduler where the initial learning rate is 1e-3, step is 10 and the decay rate $\gamma = 0.1$. For MSRVTT, we set the batch size to 64 and use the Adam optimizer with a StepLR scheduler where the initial learning rate is 3e-3, step is 1 and the decay rate $\gamma = 0.96$.

Method	#Tunable	ACC-2	F1	ACC-7	MAE	Corr
Full fine-tuning	80M	82.6	82.5	33.2	0.93	0.70
LoRA (r=16) (Hu et al. 2021)	4.0M	81.2	81.0	29.3	0.96	0.67
Adapter (d=16) (Houlsby et al. 2019)	3.9M	81.1	80.9	29.4	0.97	0.66
Adapter (d=64) (Houlsby et al. 2019)	15.7M	81.6	81.6	30.1	0.95	0.68
P-tuning (Liu et al. 2023a)	5.0M	81.3	81.2	28.4	0.98	0.66
MaPLe (Khattak et al. 2023)	5.2M	82.1	82.0	31.2	0.95	0.68
PMF (Li et al. 2023b)	4.8M	81.9	81.8	30.9	0.95	0.67
Wander (d=16)	**0.3M**	82.5	82.4	32.8	0.93	0.68
Wander (d=64)	0.9M	**83.2**	**82.9**	**33.6**	**0.92**	**0.71**

Table 3: Comparison of performance on the CMU-MOSI dataset.

Method	#Tunable	Text→Video				Video→Text			
		R@5	R@10	MdR	MnR	R@5	R@10	MdR	MnR
Full fine-tuning	134M	**57.2**	69.3	4.0	**22.4**	57.8	68.5	4.0	**20.1**
LoRA (r=16) (Hu et al. 2021)	5.8M	53.8	66.9	5.0	27.3	54.1	66.2	5.0	24.4
Adapter (d=8) (Houlsby et al. 2019)	4.6M	51.2	64.2	5.0	29.6	51.6	63.9	5.0	26.0
Adapter (d=16) (Houlsby et al. 2019)	9.2M	53.1	66.4	5.0	27.4	53.8	65.8	5.0	24.8
P-tuning (Liu et al. 2023a)	1.0M	54.1	67.3	4.0	26.9	54.8	66.9	4.0	23.9
MaPLe (Khattak et al. 2023)	1.1M	55.3	68.2	4.0	25.8	56.1	67.8	4.0	23.6
PMF (Li et al. 2023b)	1.0M	54.6	67.8	4.0	26.4	55.3	67.2	4.0	23.7
Wander (d=8)	**0.4M**	56.4	**69.4**	4.0	22.8	57.3	**68.9**	4.0	21.4
Wander (d=16)	0.8M	**57.2**	**69.4**	4.0	**22.4**	**58.0**	**68.9**	4.0	**20.1**

Table 4: Comparison of performance on the MSRVTT dataset.

Method	#Tunable	ACC-2	F1	ACC-7	MAE	Corr
SF-OP	3.5M	81.6	81.2	29.9	0.96	0.67
SF-VF	1.2M	81.5	81.3	29.6	0.96	0.67
Wander (d=16)	**0.1M**	81.8	81.4	30.6	0.95	0.67
−nonlinearity	0.2M	81.6	81.3	29.8	0.96	0.67
−residual	0.1M	81.5	81.2	29.6	0.96	0.67
Wander (d=32)	0.1M	**82.2**	**82.1**	**31.0**	**0.94**	**0.68**
−nonlinearity	0.2M	82.1	82.1	31.1	0.94	0.68
−residual	0.2M	81.9	81.7	30.9	0.94	0.68

Table 5: Comparison of Wander and its two original forms on CMU-MOSI. −c denotes we discard c in Wander.

4.2 Comparison with State-of-the-arts

For UPMC-Food 101 which has only two modalities, we compare Wander with state-of-the-art efficient transfer learning methods, including unimodal methods LoRA (Hu et al. 2021), Adapter (Houlsby et al. 2019), P-tuning (Liu et al. 2023a) and vision-language transfer learning methods MaPLe (Khattak et al. 2023), UniAdapter (Lu et al. 2024), PMF (Li et al. 2023b) and Aurora (Wang et al. 2024). Particularly, UniAdapter is based on BLIP (Li et al. 2022). Therefore, we modify it slightly for our backbones. For datasets with more modalities, we compare Wander with LoRA (Hu et al. 2021), Adapter (Houlsby et al. 2019), P-tuning (Liu et al. 2023a), MaPLe (Khattak et al. 2023) and PMF (Li et al. 2023b). We modify the prompts design of MaPLe and PMF to fit more modality situations. For MaPLe, we use a cascade structure to condition the prompts between modalities.

Table 1, 2, 3 and 4 presents the results on the four datasets with different numbers of modalities. We can observe that Wander consistently outperforms other efficient transfer learning methods and can achieve competitive results as full fine-tuning models. Besides, Wander has fewer parameters than other methods. Particularly, on the CMU-MOSI and MSRVTT datasets which have three and seven modalities, Wander outperforms the full fine-tuning models, indicating its superiority on situations where there are more than two modalities. When trained with more tunable parameters (*i.e.* the larger down projection dimension), Wander can further boost the performance. The results in the tables demonstrate that Wander can enable sufficient interactions between modalities in a token-level and parameter-efficient way.

4.3 Effectiveness of Wander

To evaluate the efficiency and effectiveness of Wander, we compare Wander with its original forms. Specifically, we denote the outer product form of sequence fusion in Equation 10 as SF-OP and the vector fusion form of sequence fusion in Equation 8 as SF-VF. The results on the CMU-MOSI dataset of Wander and its two original forms are presented in Table 5. We reduce the dimension of the features because the weight matrices $\mathbf{W}_h \in \mathbb{R}^{d_1 \times d_2 \times \cdots \times d_M \times d_h}$ and $\mathbf{W}_t \in \mathbb{R}^{d_t \times \ell_1 \times \ell_2 \times \cdots \times \ell_M}$ in SF-OP are very high-dimensional matrices. From the table, we can observe that SF (row 4) can rival its original forms SF-OP and SF-VF. This indicates the effectiveness of the low-rank decomposition of the high-dimensional matrices $\mathbf{W}_h$ and $\mathbf{W}_t$ without performance degradation. Moreover, Wander (d=16) and Wander (d=32) both outperform SF-OP and SF-VF, indicating the effectiveness of the design of Wander. SF-OP and SF-VF are both linear operations whereas in Wander we add nonlinearity and residual block to further enhance the performance.

Method	GPU time (s)	Memory (MB)	FLOPs
SF-OP	9.16	3300	0.003
SF-VF	7.87	622	0.002
SF	**0.015**	**348**	**0.002**

Table 6: Computational analysis of low-rank sequence fusion and its two original forms on CMU-MOSI.

Method	ACC-2	F1	ACC-7	MAE	Corr
Wander (VF)	80.9	80.4	29.3	0.98	0.67
Wander (SF)	**83.2**	**82.9**	**33.6**	**0.92**	**0.71**

Table 7: Benefits of sequence fusion on CMU-MOSI. VF denotes the vector fusion and SF denotes the sequence fusion. The down-projection dimensions are both 64.

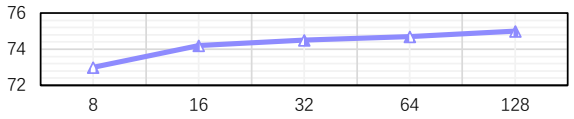


Figure 4: The impact of d on the performance on IEMOCAP.

4.4 Cost Analysis

For the efficiency of Wander, from Table 5, because we reduce the dimension of the features to enable the training of SF-OP and SF-VF, we fail to observe a substantial reduction in the number of parameters. However, mathematically, according to Equation 10, it is easy to observe that the complexity of the number of parameters of SF-OP is $\mathcal{O}(d_h \times \prod_{i=1}^{m} d_m + d_t \times \prod_{i=1}^{m} \ell_m)$. Similarly, the complexity of SF-VF is $\mathcal{O}(R_h \times d_h \times \sum_{i=1}^{m} d_m + d_t \times \prod_{i=1}^{m} \ell_m)$ and the complexity of Wander is reduced to $\mathcal{O}(R_h \times d_h \times \sum_{i=1}^{m} d_m + R_t \times d_t \times \sum_{i=1}^{m} \ell_m)$. For example, if there are three modalities and the shape of each modality is $(10, 768)$ (*i.e.* the length is 10 and the dimension is 768), then the number of parameters of SF-OP is around 348B while that of Wander is only around 14M. This fully demonstrates the efficiency of Wander. Additionally, we present the comparison of GPU time, memory and FLOPs in Table 6. We use a batch size of 24 for all methods and still reduce the dimension of features. Despite the reduction in dimension, we can still observe a substantial reduction in GPU time and memory, indicating the effectiveness of our low-rank decomposition.

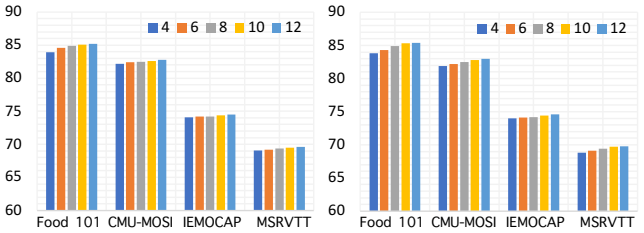
4.5 Ablation Study

Vector Fusion and Sequence Fusion. To validate the core component of Wander, we compare the vector fusion and our proposed sequence fusion in Table 7 on the CMU-MOSI dataset. Specifically, we choose the first token (CLS token) (Dosovitskiy et al. 2020) of the Transformer as the fusion vector in vector fusion. From the table, we can observe that our proposed sequence fusion significantly outperforms the vector fusion, indicating the effectiveness and superiority of our sequence fusion method.

Rank R_h and R_t. To explore the impact of the CP decomposition rank on the performance of the model, we select different ranks and present the results in Figure 5. From the figure, we can observe that with the increase of the rank, the performance of the model slightly improves. Besides, compared to R_h, the increase of R_t brings more improvement. However, the value of the rank will not affect the performance significantly, indicating the robustness of Wander.

Pre-training	Method	CMU-MOSI		IEMOCAP	
		ACC	F1	ACC	F1
HowTo100M	Full fine-tuning	82.6	82.5	74.8	74.3
	Wander(d=16)	82.5	82.4	74.2	73.8
	Wander(d=64)	83.2	82.9	74.7	74.4
CMU-MOSEI	Full fine-tuning	83.3	83.2	75.3	74.9
	Wander(d=16)	83.2	83.1	75.1	74.8
	Wander(d=64)	83.6	83.4	75.6	75.3

Table 8: The impact of the pre-training datasets on the performance of the model on CMU-MOSI and IEMOCAP datasets.



(a) The impact of R_h (b) The impact of R_t

Figure 5: The impact of the rank of CP decomposition on the performance. We report binary accuracy for UPMC-Food 101, CMU-MOSI and IEMOCAP and R@10 for MSRVTT.

Rank d. In Figure 4, we explore the impact of d on the performance of the model. As d increases, the performance improvement decreases slowly. Empirically, we can set d to around 1/8 of the original dimension. This will achieve a balance between performance and number of parameters.

Pre-training Datasets. To explore the impact of pre-training datasets on the performance of the model, we use two different datasets for pre-training and present the results on CMU-MOSI and IEMOCAP in Table 8. Specifically, CMU-MOSEI is a large multimodal sentiment analysis dataset. Therefore, we choose it as one of the pre-training datasets. From the table, we can observe that Wander can achieve good performance in both pre-training settings, indicating the universality of Wander.

5 Conclusion

In this paper, we address the two limitations of existing multimodal transfer learning methods: 1) existing methods focus on vision language transfer learning, failing to extend to situations with more modalities. 2) existing methods exhibit limited exploitation of interactions between modalities. Therefore, we propose the low-rank sequence multimodal adapter (Wander). Wander enables fine-grained token-level interactions between sequences of different modalities in a parameter-efficient way. We conduct extensive experiments on four datasets with different numbers of modalities. Wander outperforms state-of-the-art transfer learning methods consistently with fewer parameters, indicating its superiority.

References

Busso, C.; Bulut, M.; Lee, C.-C.; Kazemzadeh, E. A.; Provost, E. M.; Kim, S.; Chang, J. N.; Lee, S.; and Narayanan, S. S. 2008. IEMOCAP: interactive emotional dyadic motion capture database. *Language Resources and Evaluation*, 42: 335–359.

Chen, Y.; Qian, S.; Tang, H.; Lai, X.; Liu, Z.; Han, S.; and Jia, J. 2024. LongLoRA: Efficient Fine-tuning of Long-Context Large Language Models. In *The Twelfth International Conference on Learning Representations*.

Dettmers, T.; Pagnoni, A.; Holtzman, A.; and Zettlemoyer, L. 2024. Qlora: Efficient finetuning of quantized llms. *Advances in Neural Information Processing Systems*, 36.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Gabeur, V.; Sun, C.; Alahari, K.; and Schmid, C. 2020. Multi-modal Transformer for Video Retrieval. In *European Conference on Computer Vision (ECCV)*.

Gao, P.; Geng, S.; Zhang, R.; Ma, T.; Fang, R.; Zhang, Y.; Li, H.; and Qiao, Y. 2024. Clip-adapter: Better vision-language models with feature adapters. *International Journal of Computer Vision*, 132(2): 581–595.

Guo, Z.; Jin, T.; Chen, J.; and Zhao, Z. 2024. Classifier-guided Gradient Modulation for Enhanced Multimodal Learning. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Guo, Z.; Jin, T.; and Zhao, Z. 2024. Multimodal Prompt Learning with Missing Modalities for Sentiment Analysis and Emotion Recognition. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1726–1736.

Houlsby, N.; Giurgiu, A.; Jastrzebski, S.; Morrone, B.; De Laroussilhe, Q.; Gesmundo, A.; Attariyan, M.; and Gelly, S. 2019. Parameter-efficient transfer learning for NLP. In *International conference on machine learning*, 2790–2799. PMLR.

Hu, E. J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Joze, H. R. V.; Shaban, A.; Iuzzolino, M. L.; and Koishida, K. 2020. MMTM: Multimodal transfer module for CNN fusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 13289–13299.

Khattak, M. U.; Rasheed, H.; Maaz, M.; Khan, S.; and Khan, F. S. 2023. Maple: Multi-modal prompt learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 19113–19122.

Li, J.; Li, D.; Savarese, S.; and Hoi, S. 2023a. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, 19730–19742. PMLR.

Li, J.; Li, D.; Xiong, C.; and Hoi, S. 2022. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, 12888–12900. PMLR.

Li, X. L.; and Liang, P. 2021. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*.

Li, Y.; Quan, R.; Zhu, L.; and Yang, Y. 2023b. Efficient multimodal fusion via interactive prompting. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2604–2613.

Li, Y.; Yu, Y.; Liang, C.; Karampatziakis, N.; He, P.; Chen, W.; and Zhao, T. 2024. LoftQ: LoRA-Fine-Tuning-aware Quantization for Large Language Models. In *The Twelfth International Conference on Learning Representations*.

Liang, T.; Xie, H.; Yu, K.; Xia, Z.; Lin, Z.; Wang, Y.; Tang, T.; Wang, B.; and Tang, Z. 2022. Bevfusion: A simple and robust lidar-camera fusion framework. *Advances in Neural Information Processing Systems*, 35: 10421–10434.

Liu, H.; Li, C.; Wu, Q.; and Lee, Y. J. 2024. Visual instruction tuning. *Advances in neural information processing systems*, 36.

Liu, X.; Zheng, Y.; Du, Z.; Ding, M.; Qian, Y.; Yang, Z.; and Tang, J. 2023a. GPT understands, too. *AI Open*.

Liu, Z.; Shen, Y.; Lakshminarasimhan, V. B.; Liang, P. P.; Zadeh, A.; and Morency, L.-P. 2018. Efficient low-rank multimodal fusion with modality-specific factors. *arXiv preprint arXiv:1806.00064*.

Liu, Z.; Tang, H.; Amini, A.; Yang, X.; Mao, H.; Rus, D. L.; and Han, S. 2023b. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In *2023 IEEE international conference on robotics and automation (ICRA)*, 2774–2781. IEEE.

Lu, H.; Huo, Y.; Yang, G.; Lu, Z.; Zhan, W.; Tomizuka, M.; and Ding, M. 2024. UniAdapter: Unified Parameter-Efficient Transfer Learning for Cross-modal Modeling. In *The Twelfth International Conference on Learning Representations*.

Miech, A.; Zhukov, D.; Alayrac, J.-B.; Tapaswi, M.; Laptev, I.; and Sivic, J. 2019. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *Proceedings of the IEEE/CVF international conference on computer vision*, 2630–2640.

Pérez-Rúa, J.-M.; Vielzeuf, V.; Pateux, S.; Baccouche, M.; and Jurie, F. 2019. Mfas: Multimodal fusion architecture search. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6966–6975.

Sung, Y.-L.; Cho, J.; and Bansal, M. 2022. Vl-adapter: Parameter-efficient transfer learning for vision-and-language tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5227–5237.

Tsai, Y.-H. H.; Bai, S.; Liang, P. P.; Kolter, J. Z.; Morency, L.-P.; and Salakhutdinov, R. 2019. Multimodal transformer for unaligned multimodal language sequences. In *Proceedings of the conference. Association for computational linguistics. Meeting*, volume 2019, 6558. NIH Public Access.

Vu, T.; Lester, B.; Constant, N.; Al-Rfou, R.; and Cer, D. 2021. Spot: Better frozen model adaptation through soft prompt transfer. *arXiv preprint arXiv:2110.07904*.

Wang, H.; Yang, X.; Chang, J.; Jin, D.; Sun, J.; Zhang, S.; Luo, X.; and Tian, Q. 2024. Parameter-efficient tuning of large-scale multimodal foundation model. volume 36.

Wang, X.; Kumar, D.; Thome, N.; Cord, M.; and Precioso, F. 2015. Recipe recognition with large multimodal food dataset. In *2015 IEEE International Conference on Multimedia Expo Workshops (ICMEW)*, 1–6.

Xing, Y.; Wu, Q.; Cheng, D.; Zhang, S.; Liang, G.; Wang, P.; and Zhang, Y. 2023. Dual modality prompt tuning for vision-language pre-trained model. *IEEE Transactions on Multimedia*.

Xu, J.; Mei, T.; Yao, T.; and Rui, Y. 2016. MSR-VTT: A Large Video Description Dataset for Bridging Video and Language. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 5288–5296.

Yan, W.; Wang, Y.; Lin, W.; Guo, Z.; Zhao, Z.; and Jin, T. 2024. Low-rank Prompt Interaction for Continual Vision-Language Retrieval. In *Proceedings of the 32nd ACM International Conference on Multimedia*, 8257–8266.

Zadeh, A.; Chen, M.; Poria, S.; Cambria, E.; and Morency, L.-P. 2017. Tensor fusion network for multimodal sentiment analysis. *arXiv preprint arXiv:1707.07250*.

Zadeh, A.; Zellers, R.; Pincus, E.; and Morency, L.-P. 2016. Multimodal Sentiment Intensity Analysis in Videos: Facial Gestures and Verbal Messages. *IEEE Intelligent Systems*, 31: 82–88.

Zang, Y.; Li, W.; Zhou, K.; Huang, C.; and Loy, C. C. 2022. Unified vision and language prompt learning. *arXiv preprint arXiv:2210.07225*.

Zhang, R.; Fang, R.; Zhang, W.; Gao, P.; Li, K.; Dai, J.; Qiao, Y.; and Li, H. 2021. Tip-adapter: Training-free clip-adapter for better vision-language modeling. *arXiv preprint arXiv:2111.03930*.