南京航空航天大學
NANJING UNIVERSITY OF AERONAUTICS AND ASTRONAUTICS

Mixture-of-Experts for Open Set Domain Adaptation: A Dual-Space Detection Approach

Zhenbang Du, Jiayu An, Yunlu Tu, Jiahao Hong, and Dongrui Wu, *Fellow, IEEE*

arxiv 2023

Methods

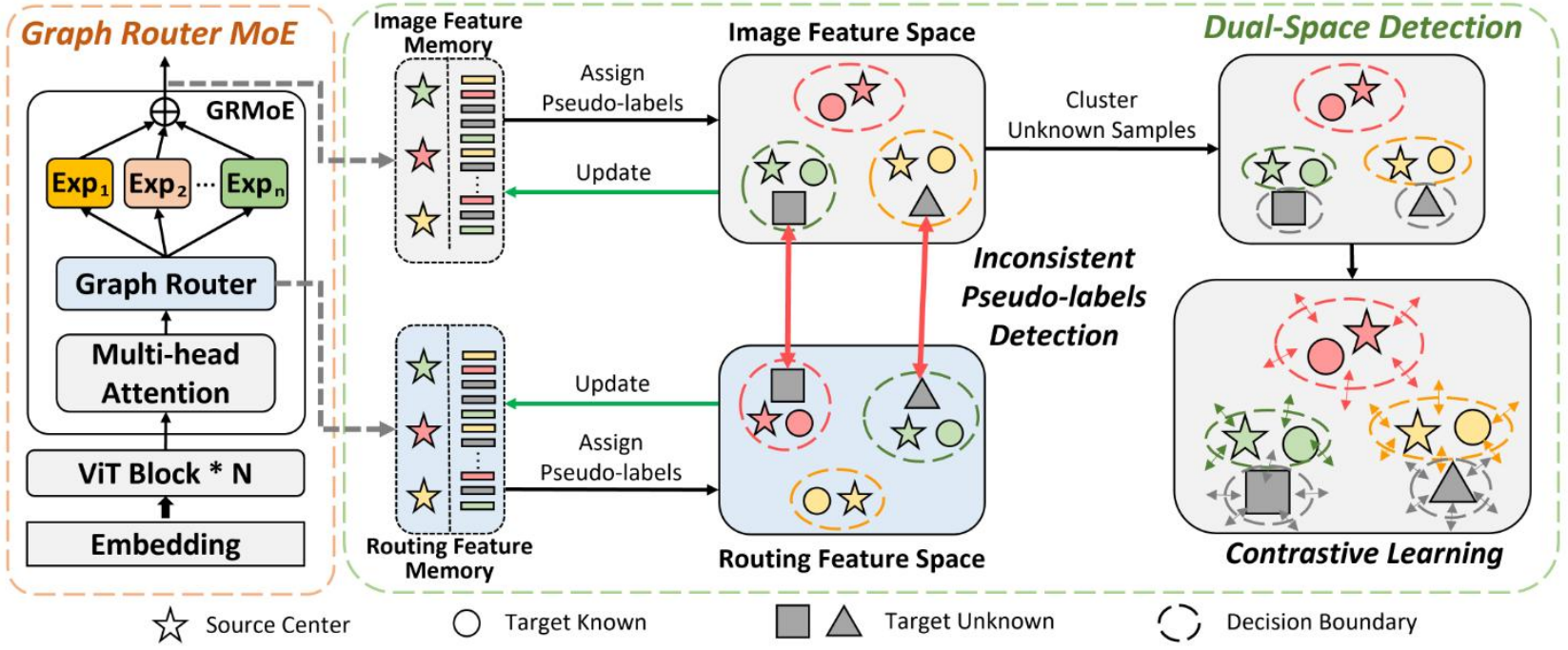


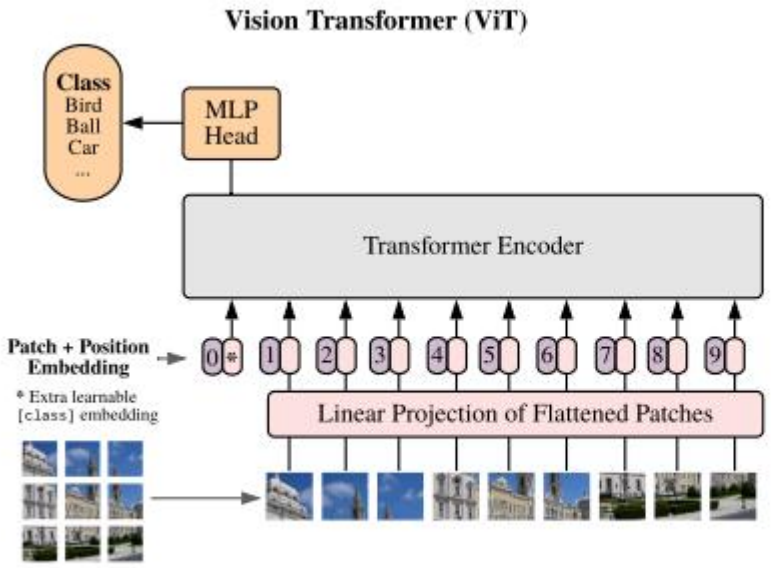
Fig. 2. Graph Router MoE (left) and DSD (right). We store the model final output image features in the image feature memory, and the routing features (note they are different from the routing scores) in the routing feature memory. We then assign pseudo-labels to target domain samples in both spaces, and those with inconsistent pseudo-labels are clustered to obtain unknown class centers. Finally, we conduct contrastive learning on all samples and update both memory banks.

- **Routing Feature**: utilize the routing feature in MoE, which has not been explored previously to the best of our knowledge.
- **Dual-Space Strategy**: a novel thresholdfree OSDA approach, to identify unknown class samples in the target domain, by exploiting the inconsistencies between the **image feature space** and the **routing feature space** in MoE.
- **Graph Router for MoE**: Graph Router, which uses a **graph neural network** as the MoE router to make better use of the spatial information in images.

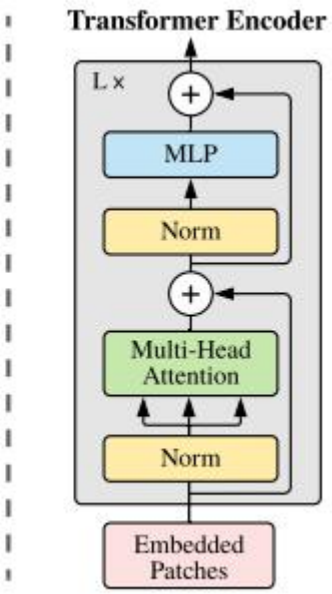
Methods

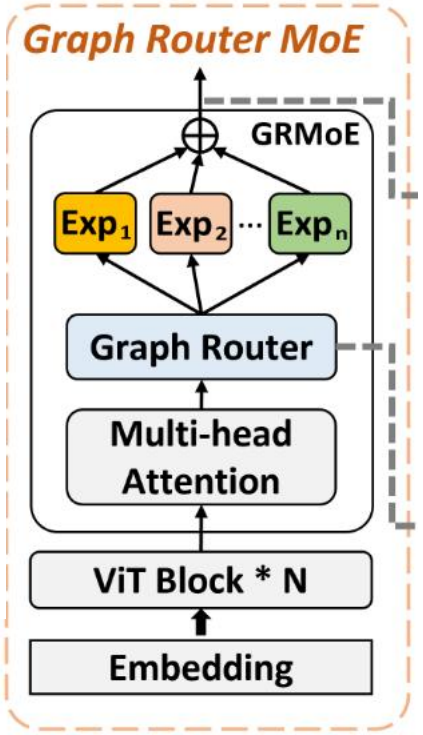
1、Graph Router MoE

The router maps samples into a routing feature space, which provides important information for identifying the unknown samples. use a graph neural network as the router to enhance the model's ability to **utilizing spatial information**.



ViT





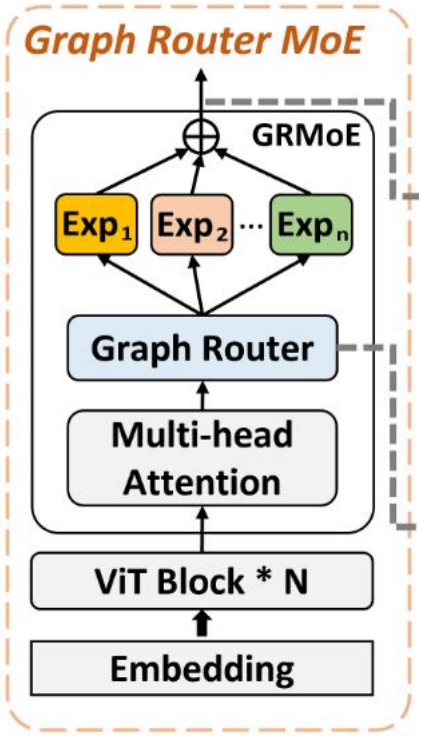
$$\mathrm{GRMoE}(\boldsymbol{x}) = \sum_{i=1}^{N} \mathrm{TOP}_K\{\mathrm{Softmax}(\mathrm{GR}(\boldsymbol{x}))\} \cdot \mathrm{EXP}_i(\boldsymbol{x}),$$

$$\mathrm{GR}(\boldsymbol{x}) = \mathrm{FC}(\mathrm{Norm}(\mathrm{GAT}(\mathcal{G}(\boldsymbol{x})))),$$

The graph is then input into the **Graph Attention Layer** GAT(·), which is normalized by Norm(·) to obtain per patch routing features.

Methods

1、Graph Router MoE



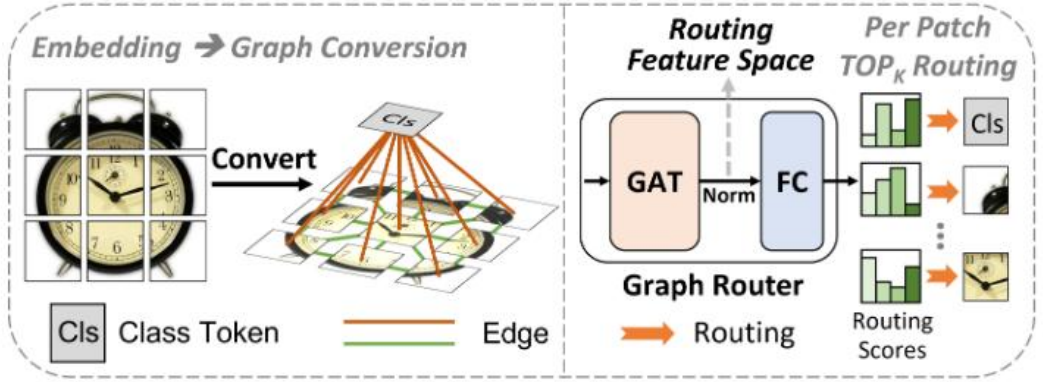


Fig. 3. Overview of the Graph Router. (Left) The conversion from the embeddings to the graph. Each patch embedding serves as a node. The edges are formed by connecting adjacent patches of the original image and linking every patch to the class token. (Right) The graph is input into the Graph Router. The routing features are extracted from the GAT layer, and the routing scores are obtained from the FC layer. 'Norm' denotes the normalization operation.

$$\mathrm{GRMoE}(\boldsymbol{x}) = \sum_{i=1}^{N} \mathrm{TOP}_K\{\mathrm{Softmax}(\mathrm{GR}(\boldsymbol{x}))\} \cdot \mathrm{EXP}_i(\boldsymbol{x}),$$

$$\mathrm{GR}(\boldsymbol{x}) = \mathrm{FC}(\mathrm{Norm}(\mathrm{GAT}(\mathcal{G}(\boldsymbol{x})))),$$

The graph is then input into the **Graph Attention Layer** GAT(·), which is normalized by Norm(·) to obtain per patch routing features.

$$\mathcal{G}(\boldsymbol{x}) = (\mathcal{V}_{\boldsymbol{x}}, \mathcal{E}_{\boldsymbol{x}}).$$

The node set Vx is the collection of the embeddings and the class token. The edge set Ex is formed by connecting adjacent patches of the original image and linking every patch to the class token.

Methods

2、Known Class Prototypes

For source domain samples Ds from known classes Cs, since **their labels are available**, the class prototypes are obtained by feeding the source samples into the model and aggregating the final image features as well as the routing features from the router.

$$\boldsymbol{c}_k^r = \frac{1}{|\mathcal{R}_k|} \sum_{\boldsymbol{r}_i \in \mathcal{R}_k} \boldsymbol{r}_i,$$
$$\boldsymbol{c}_k^f = \frac{1}{|\mathcal{F}_k|} \sum_{\boldsymbol{f}_i \in \mathcal{F}_k} \boldsymbol{f}_i,$$

where R_k and F_k denote class k's routing features and image features, respectively.

3、Momentum Memory

We introduce a momentum memory for both the image feature space and the routing feature space to **stabilize the learning**. In each iteration, the encoded feature vectors in each mini-batch are used to update the momentum memory.

$$\boldsymbol{c}_k^r \leftarrow m\boldsymbol{c}_k^r + (1-m) \cdot \frac{1}{|\mathcal{B}_k^r|} \sum_{\boldsymbol{r}_i \in \mathcal{B}_k^r} \boldsymbol{r}_i,$$
$$\boldsymbol{c}_k^f \leftarrow m\boldsymbol{c}_k^f + (1-m) \cdot \frac{1}{|\mathcal{B}_k^f|} \sum_{\boldsymbol{f}_i \in \mathcal{B}_k^f} \boldsymbol{f}_i,$$

$$\boldsymbol{r}_i^t \leftarrow m\boldsymbol{r}_i^t + (1-m)\boldsymbol{r}_{i'}^t,$$
$$\boldsymbol{f}_i^t \leftarrow m\boldsymbol{f}_i^t + (1-m)\boldsymbol{f}_{i'}^t.$$

Methods

南京航空航天大學 NANJING UNIVERSITY OF AERONAUTICS AND ASTRONAUTICS

4、Unknown Class Prototypes

For an unlabeled target sample x_j^t , we first obtain its image features f_j^t and routing features r_j^t from the memory banks, and then compute the corresponding **cosine distance** d(·, ·) to each known class prototype.

$$\hat{y}_j^r = \arg\min_k (d(\boldsymbol{r}_j^t, \boldsymbol{c}_k^r)),$$
$$\hat{y}_j^f = \arg\min_k (d(\boldsymbol{f}_j^t, \boldsymbol{c}_k^f)).$$

$$\hat{y}_j = \begin{cases} \text{unknown}, & \text{if } \hat{y}_j^r \neq \hat{y}_j^f, \\ \hat{y}_j^f, & \text{if } \hat{y}_j^r = \hat{y}_j^f. \end{cases}$$

K-Mean++

同簇内平均距离：$a(x) = \frac{1}{|\mathcal{P}_I| - 1} \sum_{x_i \in \mathcal{P}_I, x_i \neq x} d(x, x_i),$ **样本x 到同簇内其他所有样本的平均距离**

到其他簇的最近平均距离：$b(x) = \min_{J \neq I} \frac{1}{|\mathcal{P}_J|} \sum_{x_i \in \mathcal{P}_J} d(x, x_i),$ **样本x 到其他任意簇 $P_J (J \neq I)$中所有样本的平均距离里，取最小的那个。**

轮廓系数：$s(x) = \frac{b(x) - a(x)}{\max\{a(x), b(x)\}},$ **若s(x) 高，说明样本在 “自身簇内很紧密” 且 “与其他簇很疏离”，聚类效果好。**

最优簇数n_u的选取:从集合{0.25|C_S|, 0.5|C_S|, 1.0|C_S|, 2|C_S|, 3|C_S|}中，选择能让整体轮廓系数最高的$\boldsymbol{n_u}$

未知类别的原型计算: $\boldsymbol{w}_k^f = \frac{1}{|\mathcal{P}_k|} \sum_{\boldsymbol{f}_i^t \in \mathcal{P}_k} \boldsymbol{f}_i^t, \quad k = 1, \ldots, n_u,$

Methods

5、Loss

where Lcon is the **contrastive loss** which drives each sample closer to the prototype of its most similar class and further away from the prototypes of the others, L_{blc} a **regularization term** to encourage balanced use of experts in MoE , and γ a hyper-parameter.

$$\mathcal{L} = \mathcal{L}_{\text{con}} + \gamma \mathcal{L}_{\text{blc}},$$

$$\mathcal{L}_{\text{con}} = -\log \frac{\exp\left(\langle \boldsymbol{f}, \boldsymbol{z}^{+} \rangle\right)}{\sum_{k=1}^{|C_s|} \exp\left(\langle \boldsymbol{f}, \boldsymbol{c}_k^{f} \rangle\right) + \sum_{k=1}^{n_u} \exp\left(\langle \boldsymbol{f}, \boldsymbol{w}_k^{f} \rangle\right)},$$

$$\mathcal{L}_{\text{blc}} = \frac{1}{2}(\mathcal{L}_{\text{imp}} + \mathcal{L}_{\text{load}}).$$

$$\text{imp}_i(X) = \sum_{\boldsymbol{x} \in X} \text{GR}_i(\boldsymbol{x}).$$

$$p_i(x) = 1 - \Phi\left(\frac{\eta_K - \text{Softmax}_i(\text{GR}(\boldsymbol{x}))}{\sigma}\right),$$

$$\text{load}_i(X) = \sum_{\boldsymbol{x} \in X} p_i(\boldsymbol{x}).$$

变异系数小，说明所有专家的“重要性”差异小，路由权重更均匀分布

$$\mathcal{L}_{\text{imp}} = \left(\frac{\text{Std}(\{\text{imp}_i(X)\}_{i=1}^{N})}{\text{Mean}(\{\text{imp}_i(X)\}_{i=1}^{N})}\right)^2.$$

鼓励路由权重在专家间均匀分布

$$\mathcal{L}_{\text{load}} = \left(\frac{\text{Std}(\{\text{load}_i(X)\}_{i=1}^{N})}{\text{Mean}(\{\text{load}_i(X)\}_{i=1}^{N})}\right)^2.$$

专家“实际被选择次数”的均衡程度

Experiments

➢ Datasets

1、**Office31**：which contains 4,652 images from 31 classes in three different domains: Amazon, Webcam, and DSLR

2、**OfficeHome**：which contains 15,500 images from 65 classes in four different domains: Art, Clipart, Product, and Real-World.

3、**VisDA**：which contains over 200,000 images from 12 classes in different domains.

$|C_s|/|y_{unk}|$ class split was 10/11 on Office31, 25/40 on OfficeHome, and 6/6 on VisDA, following previous studies

➢ Evaluation metric

HOS：which calculates the harmonic mean（调和平均值） of the average accuracies of known classes (OS*) and the accuracy of unknown class (UNK):

$$\mathrm{HOS} = \frac{2\mathrm{OS}^* \times \mathrm{UNK}}{\mathrm{OS}^* + \mathrm{UNK}},$$

Experiments

南京航空航天大學 NANJING UNIVERSITY OF AERONAUTICS AND ASTRONAUTICS

TABLE I

RESULTS (%) ON **OFFICE31**. BEST AVERAGE HOS IN **BOLD** AND SECOND BEST WITH AN UNDERLINE.

Approach	Amazon→DSLR			Amazon→Webcam			DSLR→Amazon			DSLR→Webcam			Webcam→Amazon			Webcam→DSLR			Avg		
	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS
OSBP	89.7	66.3	76.2	89.1	66.2	75.9	62.1	78.8	69.4	79.8	86.2	82.9	76.9	66.9	71.6	93.6	77.1	84.6	81.9	73.6	76.8
STA	97.3	75.4	85.0	99.0	71.7	83.2	91.5	70.7	79.7	99.7	61.0	75.7	93.4	65.4	77.0	100.0	55.4	71.3	96.8	66.6	78.7
ROS	66.5	86.3	75.1	38.4	71.7	50.0	66.3	79.8	72.4	87.9	82.2	90.0	66.5	87.0	75.4	95.1	77.7	85.5	70.1	80.8	74.7
PGL	80.1	65.1	71.8	90.1	68.3	77.7	62.0	58.0	60.0	87.1	65.3	74.6	69.6	59.2	64.0	77.3	61.8	68.7	77.7	63.0	69.5
DCC	98.7	89.8	94.0	93.9	92.8	93.3	94.5	75.2	83.8	100.0	92.8	96.3	95.2	81.2	87.6	100.0	89.2	94.3	97.1	86.8	91.6
DANCE	84.1	33.1	47.5	91.5	55.4	69.0	67.5	63.6	65.5	97.9	55.0	70.4	92.0	47.4	62.5	100.0	48.0	64.9	88.8	50.4	63.3
OVANet	92.3	89.7	91.0	91.6	91.8	91.7	51.7	99.3	68.0	96.9	100.0	98.4	86.1	96.3	90.9	100.0	85.2	92.0	86.4	93.7	88.7
GATE†	-	-	88.4	-	-	86.5	-	-	84.2	-	-	95.0	-	-	86.1	-	-	96.7	-	-	89.5
UADAL	81.2	88.6	84.7	84.3	96.7	90.1	72.6	90.0	80.4	100.0	92.5	96.1	72.8	92.4	81.4	100.0	98.0	99.0	85.2	93.0	88.6
ANNA	94.0	73.4	82.4	94.6	71.5	81.5	73.7	82.1	77.7	99.5	97.0	98.2	73.7	82.6	77.9	100.0	89.9	94.7	89.3	82.8	85.4
GLC	85.3	90.6	87.9	86.8	93.1	89.8	92.3	98.0	95.1	94.0	96.4	95.2	91.9	97.9	94.8	98.7	96.9	97.8	91.5	95.5	93.4
DSD (Ours)	96.9	91.3	94.0	91.2	94.4	92.8	91.5	97.0	94.1	97.9	91.9	94.8	93.4	95.3	94.4	99.1	91.3	95.0	95.0	93.5	**94.2**

† Cited from [42].

TABLE II

RESULTS (%) ON **OFFICEHOME**. 'THRESHOLD-FREE' MEANS NO THRESHOLD IS REQUIRED. BEST AVERAGE HOS IN **BOLD** AND SECOND BEST WITH AN UNDERLINE.

Approach	Threshold-Free	Art→Clipart			Art→Product			Art→Real-World			Clipart→Art			Clipart→Product			Clipart→Real-World		
		OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS
OSBP	✗	46.8	77.2	58.3	55.3	72.0	62.6	75.8	55.6	64.2	52.6	61.8	56.8	59.2	47.1	52.5	67.7	63.5	65.5
STA	✓	60.0	56.2	58.0	83.1	47.1	60.1	90.6	49.2	63.8	68.9	63.0	65.8	74.5	49.4	59.4	77.7	51.4	61.9
ROS	✗	48.7	73.0	58.4	64.1	66.0	65.0	73.8	62.6	67.7	52.6	64.4	57.9	59.8	53.9	56.7	58.2	76.0	66.0
PGL	✗	63.8	51.3	56.9	80.2	58.4	67.6	88.7	63.4	73.9	71.0	59.0	64.4	74.1	56.4	64.1	81.4	59.5	68.8
DCC	✓	56.7	69.3	62.3	78.9	67.2	72.6	82.2	66.8	73.7	54.1	56.1	55.1	67.8	73.9	70.7	82.7	76.6	79.5
DANCE	✗	44.7	65.1	53.0	60.3	60.0	60.1	80.7	68.9	74.3	45.6	74.7	56.6	64.5	68.7	66.5	39.7	86.7	54.4
OVANet	✓	42.6	81.4	55.9	72.3	65.6	68.8	86.1	64.1	73.5	48.7	83.4	61.5	63.1	73.9	68.1	70.1	78.1	73.9
GATE†	✓	-	-	63.8	-	-	70.5	-	-	75.8	-	-	66.4	-	-	67.9	-	-	71.7
UADAL	✗	42.6	71.3	53.4	68.6	74.6	71.5	85.7	73.2	78.9	52.1	82.5	63.9	59.9	74.8	66.5	65.4	83.7	73.4
ANNA	✗	58.5	72.3	64.7	67.6	70.8	69.2	72.7	75.3	74.0	48.9	82.5	61.4	61.7	69.9	65.6	68.4	76.4	72.2
GLC	✗	62.6	69.3	65.8	75.6	73.6	74.6	81.4	80.2	80.8	71.4	34.4	46.4	77.9	76.2	77.0	82.1	82.2	82.1
DSD (Ours)	✓	49.5	78.1	60.6	63.6	78.7	70.3	82.3	75.6	78.8	65.4	69.4	67.3	62.5	82.6	71.2	71.8	79.9	75.6

Approach	Product→Art			Product→Clipart			Product→Real-World			Real-World→Art			Real-World→Clipart			Real-World→Product			Avg		
	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS
OSBP	53.2	64.5	58.3	49.6	45.2	47.3	63.3	70.1	66.6	65.8	43.1	52.1	59.2	34.1	43.2	80.3	51.8	63.0	60.7	57.2	57.5
STA	70.2	62.3	66.0	57.3	54.8	56.0	86.2	57.7	69.2	77.9	53.4	63.4	59.5	43.4	50.2	86.3	41.1	55.7	74.4	52.4	60.8
ROS	48.5	63.9	55.1	44.3	73.5	55.3	66.3	83.3	73.8	68.1	59.9	63.7	48.5	71.9	57.9	71.3	46.2	56.1	58.7	66.2	61.1
PGL	72.0	60.6	65.8	63.2	52.3	57.2	82.5	60.3	69.7	73.7	59.2	65.7	63.0	51.0	56.4	83.6	60.1	69.9	74.8	57.6	65.0
DCC	55.1	78.6	64.8	51.7	73.1	60.6	76.6	73.9	75.2	75.5	55.9	64.2	57.4	65.0	60.9	79.2	74.0	76.5	68.2	69.2	68.0
DANCE	59.7	54.0	56.7	51.3	63.4	56.7	74.8	66.1	70.1	72.4	30.0	42.5	64.8	30.9	41.8	81.1	43.7	56.8	61.6	59.4	57.5
OVANet	50.1	84.8	63.0	37.3	83.5	51.6	77.4	74.8	76.1	72.3	69.0	70.6	46.5	70.7	56.1	82.5	56.1	66.8	62.4	73.8	65.5
GATE†	-	-	67.3	-	-	61.5	-	-	76.0	-	-	70.4	-	-	61.8	-	-	75.1	-	-	69.0
UADAL	59.6	81.3	68.8	32.5	75.1	45.3	79.4	78.8	79.1	73.6	70.7	72.2	38.3	73.9	50.5	80.8	70.8	75.5	61.5	75.9	66.6
ANNA	52.0	80.9	63.3	51.1	79.6	62.2	67.8	79.6	73.2	58.8	83.5	69.0	56.1	75.7	64.4	73.0	85.6	78.8	61.4	77.7	68.2
GLC	61.5	84.5	71.2	48.5	91.6	63.4	75.1	78.2	76.6	75.9	35.4	48.2	36.1	90.2	51.6	83.4	81.2	82.3	69.3	73.1	68.3
DSD (Ours)	61.5	82.1	70.3	42.5	83.8	56.4	78.8	76.6	77.7	73.7	71.2	72.5	47.0	80.1	59.3	76.1	79.8	77.9	64.6	78.2	**69.8**

Experiments

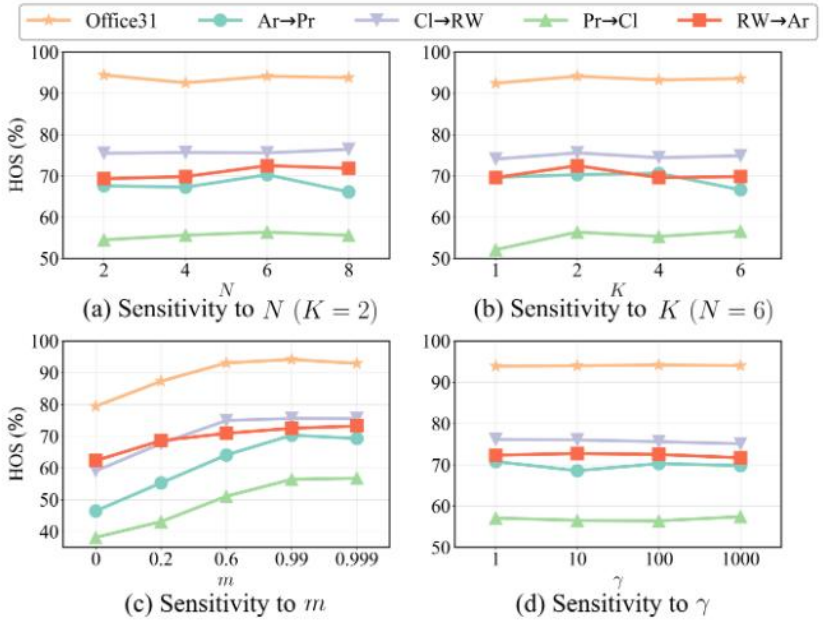


Fig. 4. Hyper-parameters analysis on Office31 and Art→Product, Clipart→Real-World, Product→Clipart and Real-World→Art on OfficeHome. (a) N, the total number of experts; (b) K, the number of experts selected during each routing step; (c) m in Eq. (8); and, (d) γ.

N： the total number of experts.
K: the number of experts selected during each routing step.
m: momentum update.
γ: loss function.

TABLE IV
COMPARISON OF DIFFERENT ROUTERS' HOS (%) ON THE THREE DATASETS.

Router	Office31	OfficeHome	VisDA
Cosine Router	93.2	67.3	72.4
Graph Router (MHSA)	76.9	60.7	67.5
Graph Router (GCN)	92.1	67.6	74.4
Graph Router (Ours)	**94.2**	**69.8**	**75.5**

replace the GAT in Graph Router with MHSA and graph convolution layer (GCN)

TABLE V
HOS (%) ON OFFICE31, OFFICEHOME, AND VISDA WITH DIFFERENT MOE SETTINGS.

Setting	Office31	OfficeHome	VisDA
Last Layer	92.7	68.6	69.3
9-th Routing	84.3	61.8	62.6
Our Setting	**94.2**	**69.8**	**75.5**

the 9-th and 11-th layers of the original 12 layers Deit-S were replaced by GRMoE layers, and the routing feature space was obtained from the 11-th layer's router output. including Last Layer (only replacing the 12-th layer of Deit-S by GRMoE) and 9-th Routing (the output of the 9-th GRMoE layer as the routing feature space).

Experiments

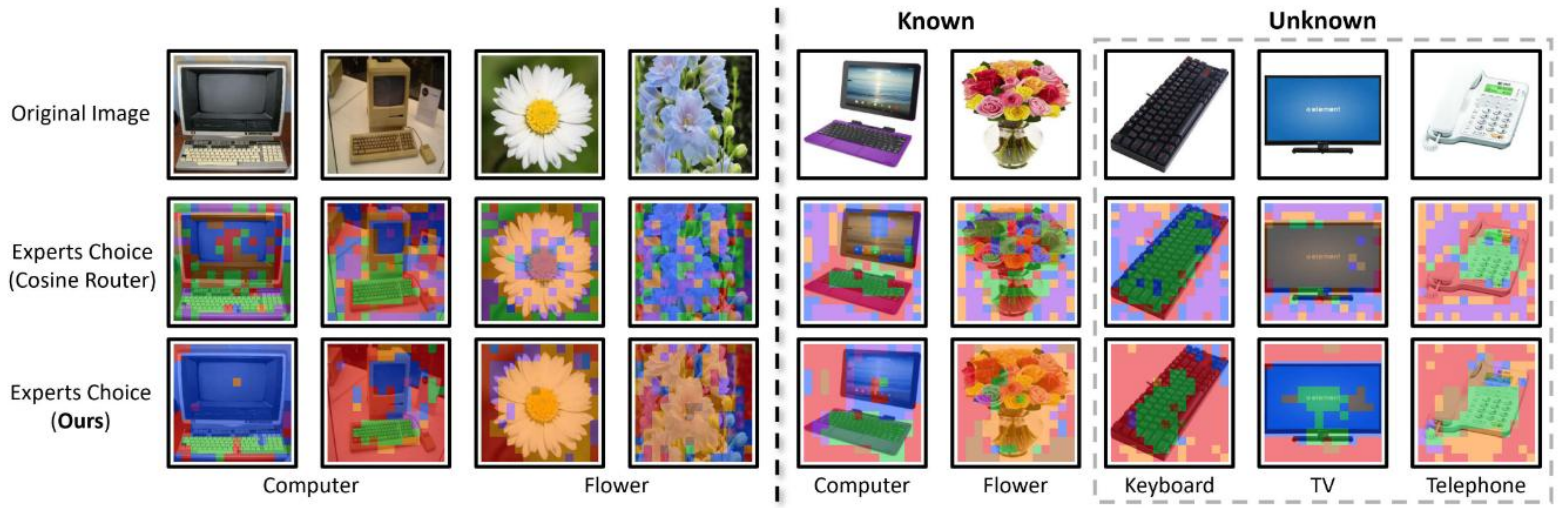


Fig. 6. Visualization of the experts choice by Cosine Router and Graph Router on OfficeHome. Left: samples from the source domain (Real-World); Right: samples from the target domain (Product). Different colors represent different dominate experts of different patches. Gray dashed borders indicate unknown class samples.

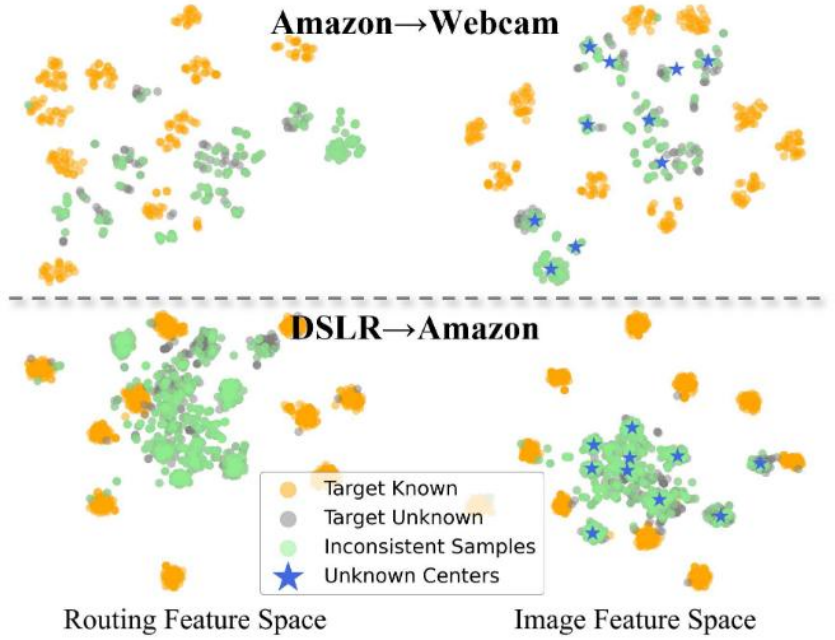
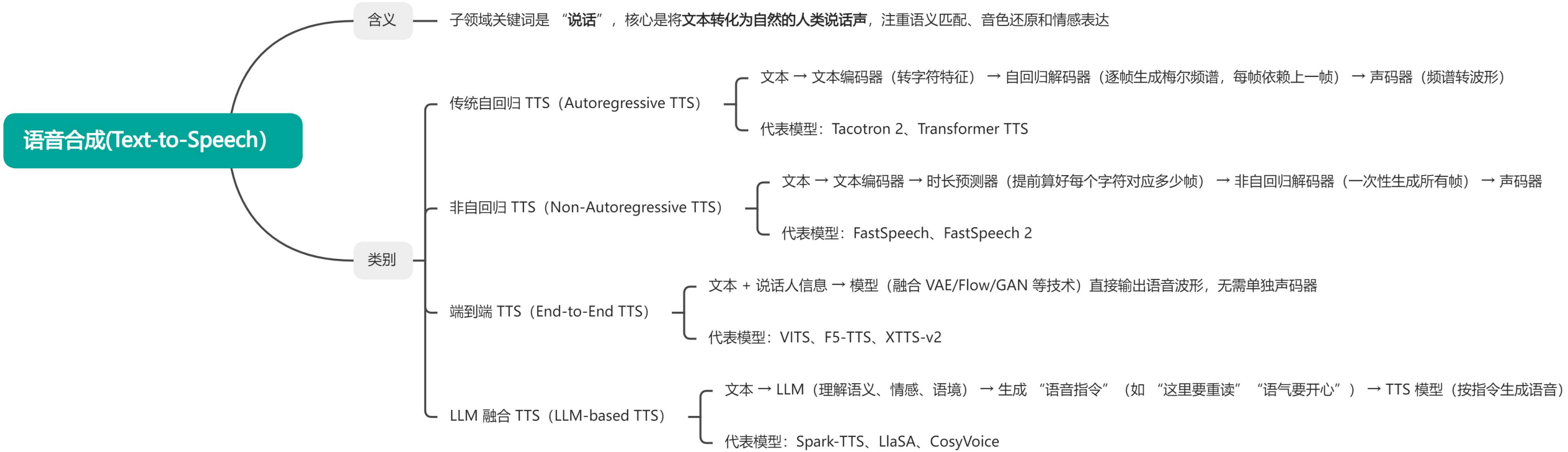


Fig. 7. T-SNE visualization of the routing feature space (left) and the image feature space (right) in Amazon→Webcam (top) and DSLR→Amazon (bottom) on Office31.

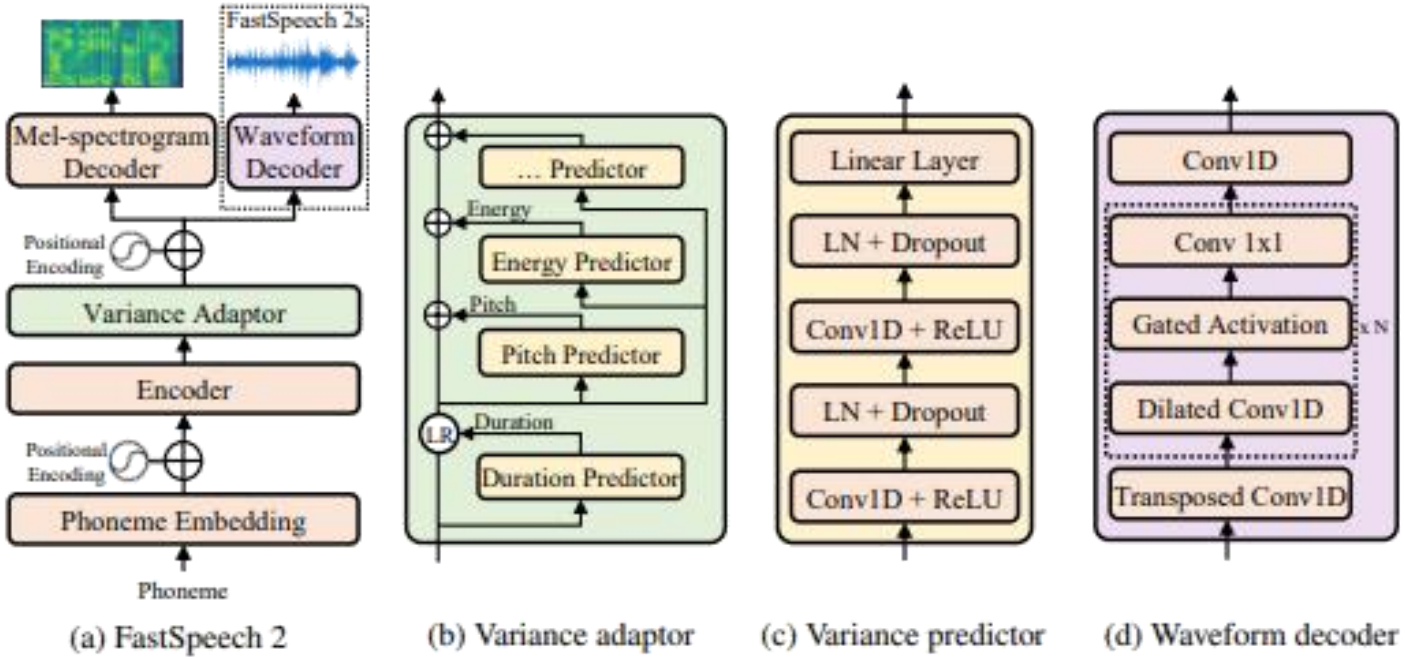
Thanks



①回归模型的本质是逐步生成输出，后一步依赖前一步结果”；非回归模型则是“并行生成所有输出，各步无依赖”。
判断标准只有一个：**输出序列的生成是否依赖历史输出。**

②端到端模型的本质是跳过人工设计的中间子模块，直接实现‘原始输入→目标输出’的映射”，无需人工拆分流程。
判断标准：是否需要**人工拆分 “输入→输出” 的中间步骤，是否依赖人工设计的子模块**

端到端+非回归模型FastSpeech



(a) FastSpeech 2　(b) Variance adaptor　(c) Variance predictor　(d) Waveform decoder

从古至今，庸君最怕**朝纲了**，可**明**君他就不怕，不但不怕，反能**借助**。要我说，你就让**李四张三**互相争宠，只要你心里**清楚**，左右**周旋**，你就能**处**于不败之**境**。

核心模块

1、**Encoder：**将输入的音素序列（Phoneme Embedding）进行处理，通过位置编码（Positional Encoding）加入位置信息，使得模型能够区分文本中不同位置的音素。

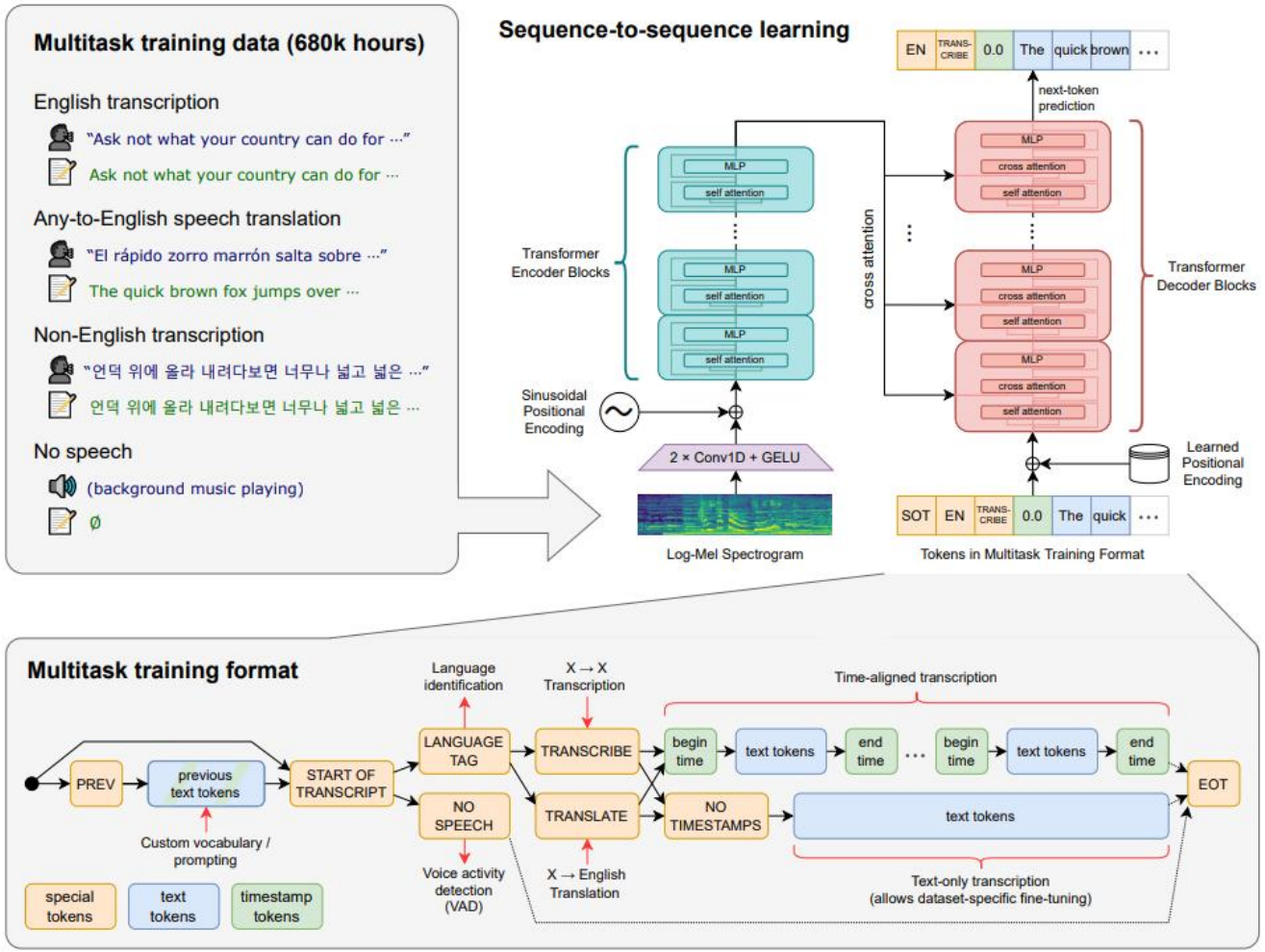
2、**Variance Adaptor：**包含多个预测器，如能量预测器、基频预测器和时长预测器。时长预测器根据编码器输出的隐藏特征预测每个音素的**持续时长**，能量预测器预测语音的能量，基频预测器预测语音的**基频（音高）。**

3、**梅尔谱解码器（Mel - spectrogram Decoder）**：接收方差适配器调整后的隐藏特征，将其转换为梅尔频谱。梅尔频谱是一种常用的语音特征表示，它模拟了人类听觉系统对不同频率声音的感知特性。

4、**波形解码器（Waveform Decoder）**：以梅尔频谱作为输入，通过一系列的卷积操作和门控激活函数，将梅尔频谱转换为语音波形。

[1]Ren Y, Ruan Y, Tan X, et al. Fastspeech: Fast, robust and controllable text to speech[J]. Advances in neural information processing systems, 2019, 32.

自动语音识别（Automatic Speech Recognition，ASR）的核心是让机器“听懂”人类**语音并转化为文本**，其实现过程本质是将连续的语音信号映射为离散的文本序列。



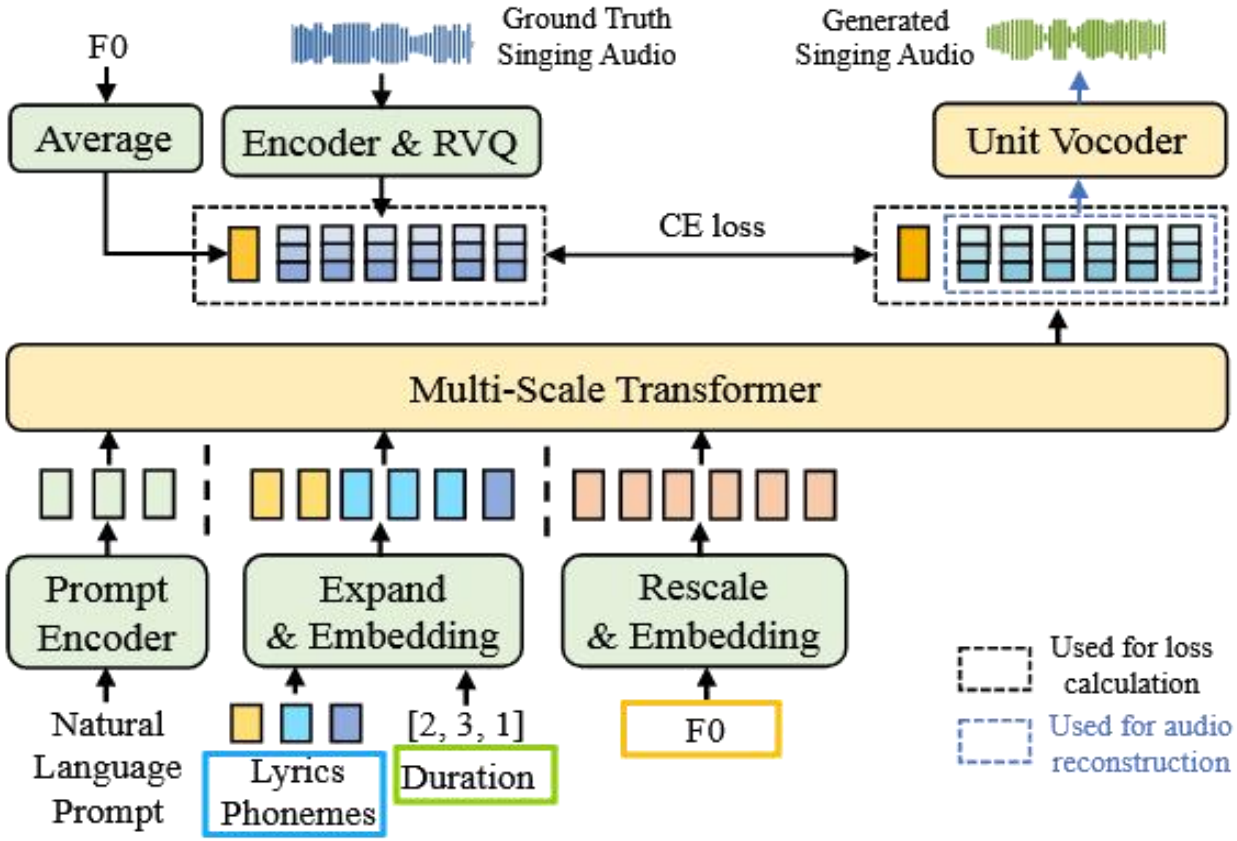
- PREV：关联“前序文本 token”，辅助模型建模上下文（如转录长句时，参考前面的词生成后面的词，保证文本连贯性）；
- LANGUAGE TAG：标记语言（如EN表示英语，KO表示韩语）；
- TRANSCRIBE：表示 “语音转文字” 任务；
- TRANSLATE：表示 “语音翻译” 任务；
- NO SPEECH：表示 “无语音” 任务。

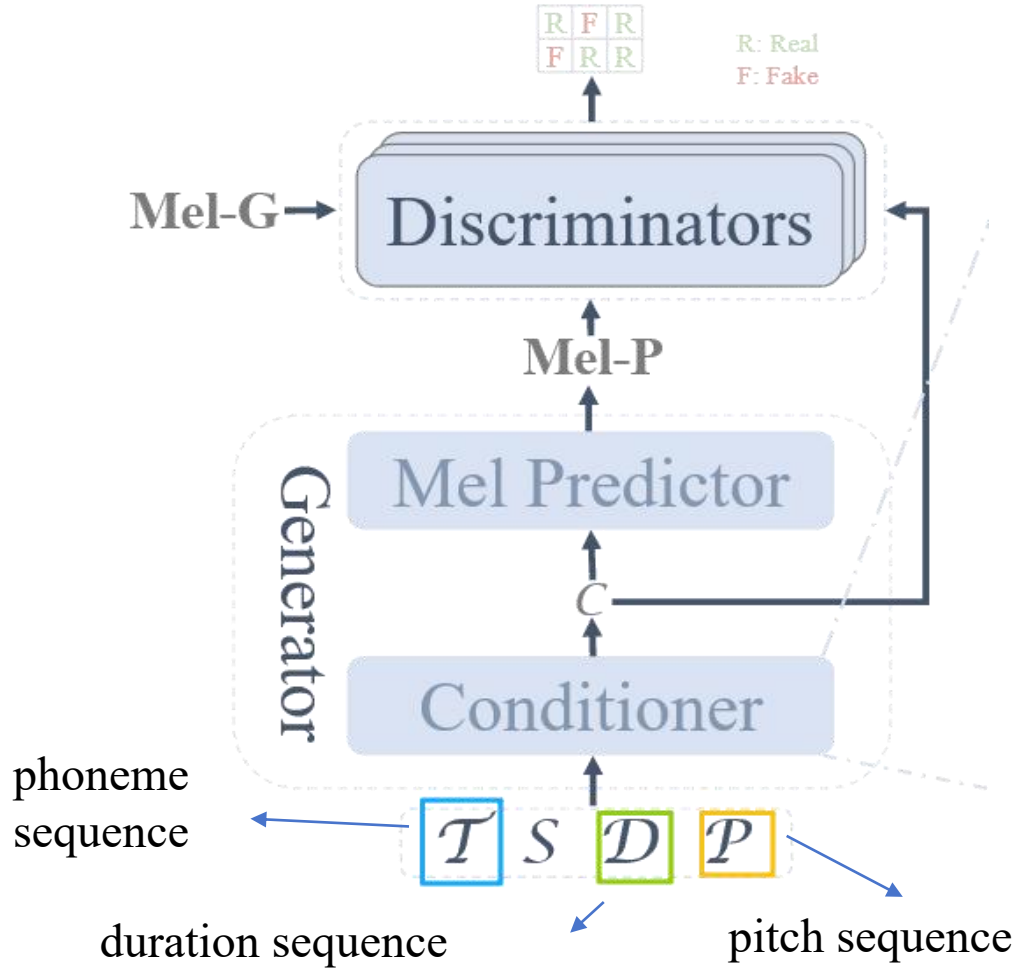
现有的模型：FireRedASR、Qwen3-ASR-Flash（支持多语种、多方言）、**Dolphin**（支持东方40语种+中国22种方言）

[1]https://openai.com/zh-Hans-CN/index/whisper/
[2]Meng Y, Li J, Lin G, et al. Dolphin: A Large-Scale Automatic Speech Recognition Model for Eastern Languages[J]. arXiv preprint arXiv:2503.20212, 2025.

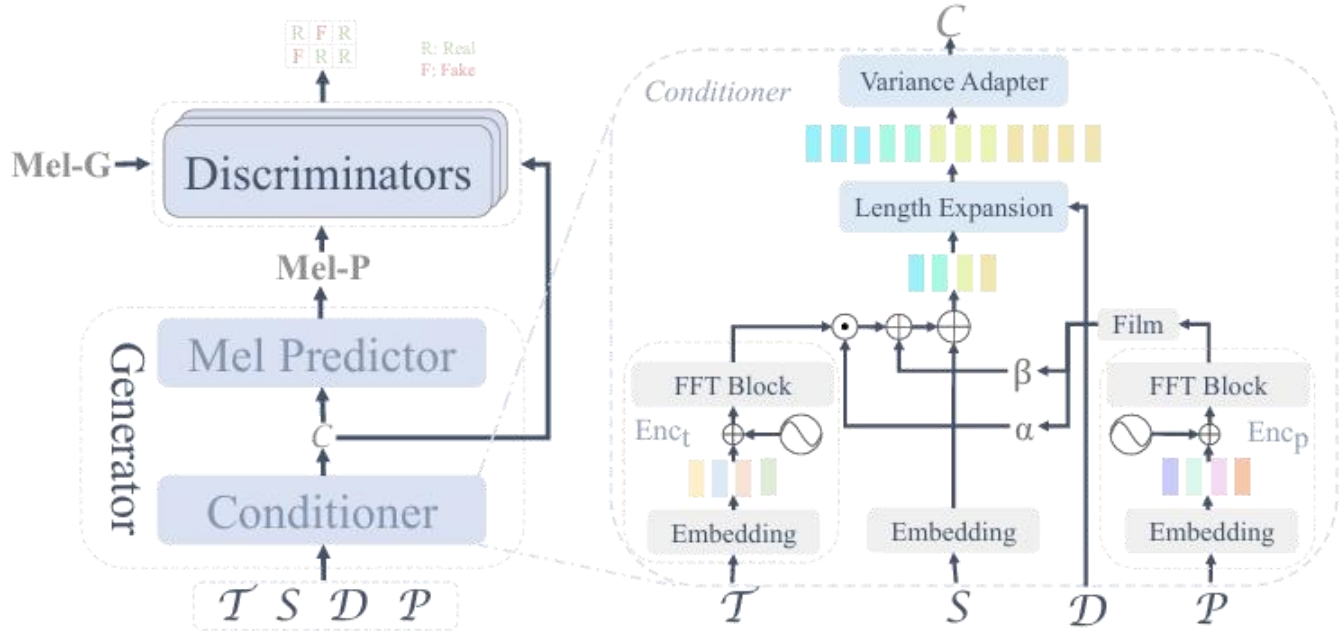
歌声合成（Singing Voice Synthesis, SVS）是将乐谱（**音符、歌词、节奏**等）转换为自然歌声的技术。其核心挑战是同时保证音高准确性（符合乐谱）、节奏匹配性（贴合节拍）和情感表现力（如欢快、悲伤）

Lyrics: 快乐缺点勇气，浪漫缺点诗意，沉默一句一句都是谜题　　ground truth　　low ptich





[1]Wang Y, Hu R, Huang R, et al. Prompt-singer: Controllable singing-voice-synthesis with natural language prompt[J]. arXiv preprint arXiv:2403.11780, 2024.
[2]Zheng M, Bai P, Shi X, et al. FT-GAN: fine-grained tune modeling for Chinese opera synthesis[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2024, 38(17): 19697-19705.



➢ **调节模块（Conditioner）**

①调节模块的输入包含四个元素：**音高序列（P）、音素序列（T）、时长序列（D）以及说话人标签（S）**

②由于歌剧音高序列中存在复杂的滑音，导致音高序列与音素序列之间的上下文信息存在显著差异，加了独立的 FFT 块（Encp）来对音高信息进行编码。

③特征调制：以更好地融合音高与音素信息。为实现特定说话人的歌剧合成。‘

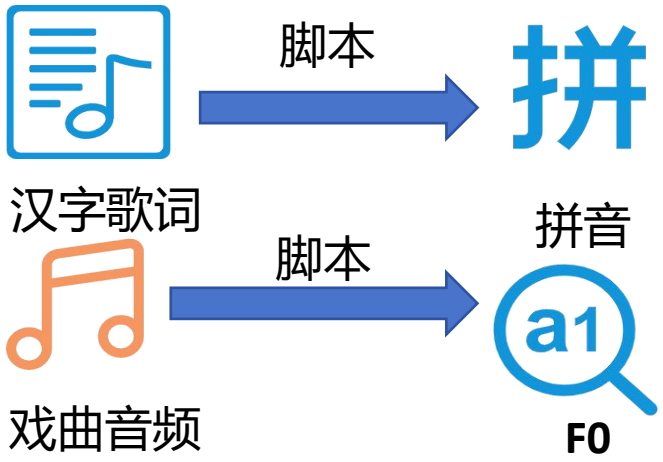
④C：统一条件表征。

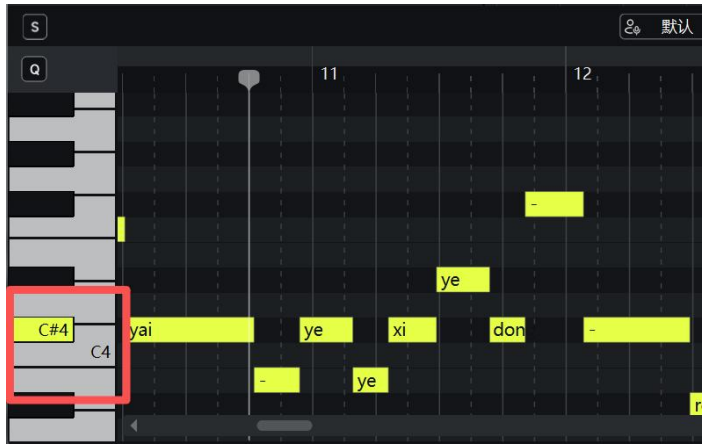
➢ **梅尔频谱预测器（Mel Predictor）**

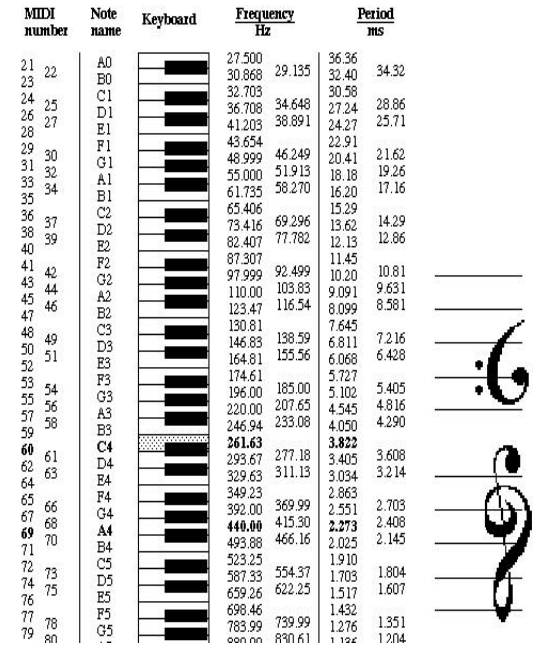
统一条件表征 C 较长且波动幅度大，既需要建模长程依赖，又要捕捉歌剧音频中显著的局部相关和平移不变性。采用 **Conformer** 作为梅尔频谱预测器的骨干网络。

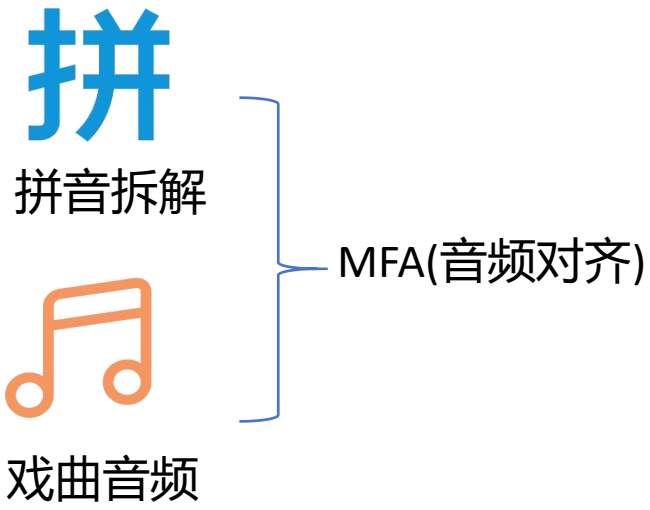
➢ **判别器（Discriminator）**

基于 **PatchGAN** 并结合统一条件表征 C 提升判别精度；为应对时长差异与音高饱满度问题，引入 **ML-GAN** 分割堆叠频谱图并改进 **SF-GAN**，采用三路加权判别器分别处理低、中、高频段，使生成音频更契合歌仔戏声学特征。










```
File type = "ooTextFile"
Object class = "TextGrid"

xmin = 0
xmax = 8.96
tiers? <exists>
size = 2
item []:
    item [1]:
        class = "IntervalTier"
        name = "words"
        xmin = 0
        xmax = 8.96
        intervals: size = 10
        intervals [1]:
            xmin = 0.0
            xmax = 0.04
            text = ""
        intervals [2]:
            xmin = 0.04
            xmax = 0.94
            text = "shi"
```