

# Double Mixture: Towards Continual Event Detection from Speech

Jingqi Kang<sup>\*,♡</sup> Tongtong Wu<sup>\*,♣</sup> Jinming Zhao<sup>♣</sup> Guitao Wang<sup>♡</sup> Yinwei Wei<sup>♣</sup> Hao Yang<sup>♣</sup>  
 Guilin Qi<sup>♡</sup> Yuan-Fang Li<sup>♣</sup> Gholamreza Haffari<sup>♣</sup>

<sup>♡</sup>Southeast University, China; <sup>♣</sup>Monash University, Australia

<sup>♡</sup>{kjg, 220222117, gqi}@seu.edu.cn,

<sup>♣</sup>{first-name.last-name}@monash.edu

## ABSTRACT

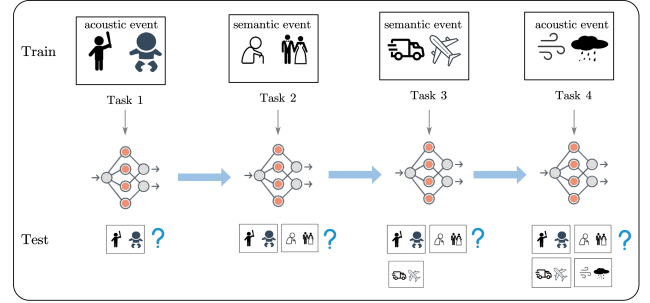
Speech event detection is crucial for multimedia retrieval, involving the tagging of both semantic and acoustic events. Traditional ASR systems often overlook the interplay between these events, focusing solely on content, even though the interpretation of dialogue can vary with environmental context. This paper tackles two primary challenges in speech event detection: the continual integration of new events without forgetting previous ones, and the disentanglement of semantic from acoustic events. We introduce a new task, continual event detection from speech, for which we also provide two benchmark datasets. To address the challenges of catastrophic forgetting and effective disentanglement, we propose a novel method, 'Double Mixture.' This method merges speech expertise with robust memory mechanisms to enhance adaptability and prevent forgetting. Our comprehensive experiments show that this task presents significant challenges that are not effectively addressed by current state-of-the-art methods in either computer vision or natural language processing. Our approach achieves the lowest rates of forgetting and the highest levels of generalization, proving robust across various continual learning sequences. Our code and data are available at <https://github.com/jodie-kang/DoubleMixture>.

## KEYWORDS

Continual Learning; Mixture of Experts; Event Detection

## 1 INTRODUCTION

Speech event detection is vital for multimedia retrieval [28], as it forms the basis for precise indexing and accessing of extensive audio content [10, 11]. This task involves recognizing not just the semantic content of dialogues, essentially the words spoken, but also the acoustic events, which indicate the location and background conditions of the speech. Current information extraction systems [11, 35] based on automatic speech recognition (ASR) systems [10] often fail to address the interaction between semantic and acoustic signals. This oversight is particularly problematic in complex multimedia



**Figure 1: In continual learning, learners incrementally acquire new event types and must evaluate all previously learned types during testing. This process is particularly challenging in speech-based scenarios due to the complex interplay of semantic content (semantic event) and background sounds (acoustic event).**

environments where background sounds and speech frequently overlap [7, 40].

To address the challenges in extracting events from real-world speech data, the models need to meet two key criteria: continual learning [2] and event disentanglement [17]. Continual learning enables models to adapt to an evolving data landscape, improving their ability to recognize new event types while retaining previously learned information. Disentangling semantic events from their acoustic environments allows models to handle rare or previously unseen event combinations effectively.

Current research typically approaches speech-based information extraction as an extension of text-based methods [4, 11, 32], focusing predominantly on end-to-end extraction from a content perspective. This approach often neglects the unique challenges of speech data. In response, we propose a new task: Continual Event Detection from Speech (CEDS). Due to a scarcity of specific datasets, we suggest utilizing existing text-based and acoustic event detection datasets to establish a new benchmark for continual learning, employing methods such as speech overlaying and speech splicing.

To tackle the challenges of continual learning and event disentanglement in speech, we introduce a novel strategy called Double Mixture. This approach combines a mixture of experts [6, 21, 43] with automatically assigning a dedicated expert to each task for accruing new knowledge, and a mixture of memory, which is a simple yet effective method for replaying speech experiences. The proposed method aims to prevent catastrophic forgetting and improve

\* denotes the equal contribution.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

ACM MM, 2024, Melbourne, Australia

© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM

<https://doi.org/10.1145/nnnnnnnn.nnnnnnnn>

the model's ability to process complex audio inputs. In summary, our contributions are as follows:

- (1) We are the first to propose the task of continual event detection from speech, providing new benchmarks and highlighting the significant challenges of catastrophic forgetting and semantic-acoustic event disentanglement.
- (2) We introduce Double Mixture, a method that merges a mixture of experts with a mixture of memory, to enhance performance in complex audio scenarios.
- (3) Our extensive experiments confirm the complexity of this task and demonstrate the state-of-the-art performance of our proposed method in preventing catastrophic forgetting and managing complex real-world audios with varying event combinations.

## 2 TASK DEFINITION

The continual speech event detection task aims to sequentially learn and recognize new tasks from a speech stream, and the process can be formally defined as follows:

Consider a task sequence  $\mathcal{T} = \{\mathcal{T}_1, \dots, \mathcal{T}_t, \dots, \mathcal{T}_T\}$ , where  $T$  denotes the sequence length. Each task  $\mathcal{T}_t = \{\mathcal{D}_t^{train}, \mathcal{D}_t^{val}, \mathcal{D}_t^{test}\}$  in the sequence consists of a supervised learning task including a training set  $\mathcal{D}_t^{train}$ , a validation set  $\mathcal{D}_t^{val}$  and a test set  $\mathcal{D}_t^{test}$ . Each dataset  $\mathcal{D}$  contains  $n$  sample pairs  $\mathcal{D} = (x_i, y_i)_{i=1}^n$ , where  $x_i$  is a sequence of digital audio signals, and  $y_i \in \mathcal{Y}$  denotes the event type. The event type can be either a semantic event or a sound event. Assuming that  $\mathcal{Y}_t^{train}$  represents the set of event types occurring in  $\mathcal{D}_t^{train}$ , the incremental speech event extraction task requires that the sets of event types do not intersect with each other during training, i.e.,  $\emptyset = \bigcap_{t=1}^T \mathcal{Y}_t^{train}$ . However, the set of event types at the test phase is  $\mathcal{Y}_T^{test} = \bigcup_{t=1}^T \mathcal{Y}_t^{test}$ , allowing the model to encounter all previously learned event types throughout the test phase of the task sequence.

The catastrophic forgetting problem is the degradation of the model's performance on the previous task after learning a new task. The event disengagement problem refers to combining the knowledge learned in previous tasks and applying it to new tasks.

## 3 METHOD

Our method employs the encoder-decoder Transformer[27], including a speech encoder and a text decoder. Inspired by the Mixed of Experts model (MoE) [21, 25], we introduce a mixture of speech expert networks in the decoder to mitigate catastrophic forgetting in continual learning. Different from the traditional experience replay strategy, we save mixed samples of semantic and sound events in memory to strengthen collaboration between different speech experts and thereby improve the generalization ability of the model.

### 3.1 Mixture of Speech Experts

The key issue in continual learning is how to avoid the loss of past knowledge while the model continues to absorb new knowledge. MoE models are shown effective in mitigating the issue in the way that they combine the strength of multiple expert systems to achieve an in-depth understanding of specific tasks and work together in the overall performance [9]. We thus introduce the mixture of speech experts in our framework where each expert focuses on a single

task in the task stream, while the overall model responds optimally to new data by dynamically adjusting the weights of each expert, and maintaining past knowledge during the learning process. As will be demonstrated in later sections, this method can improve the generalization ability of the model, i.e., it can use the learned event knowledge to identify new event combinations.

We design each speech expert as a bottleneck adapter network, which consists of a down-projection layer  $W_{down}$  for dimensionality reduction, a nonlinear activation layer  $\sigma(\cdot)$ , and an up-projection layer  $W_{up}$  for dimensionality restoration:

$$E(\mathbf{H}) = \mathbf{H} + \sigma(\mathbf{H} \times W_{down})W_{up}, \quad (1)$$

where  $\mathbf{H} \in \mathbb{R}^{(p \times d)}$  is the hidden layer representation from the  $l$ -th layer, where  $d$  is the dimension of the hidden layer and  $p$  is the sequence length.  $W_{down} \in \mathbb{R}^{d \times b}$  and  $W_{up} \in \mathbb{R}^{b \times d}$  are the trainable parameters of the adapter layer, and  $b$  is the bottleneck dimension.

The core function of the adapter is to enhance the model's ability to process specific features, which is achieved by adding a learnable transformation to the original input:

$$A(\mathbf{x}) = \sigma(W_f \times \mathbf{x} + b_f) + \mathbf{x}, \quad (2)$$

where  $A(\mathbf{x})$  represents the output of the adapter,  $\mathbf{x}$  is the input feature,  $W_f$  and  $b_f$  are the weights and biases of the adapter, and  $\sigma$  denotes the activation function.

These weights  $\alpha_i$  are usually normalized to ensure that the weights of all the experts sum to one, thus keeping the size of the output constant:

$$\alpha_i = \frac{e^{G_i(\mathbf{x})}}{\sum_{j=1}^n e^{G_j(\mathbf{x})}}, \quad (3)$$

where  $G_i(\mathbf{x})$  is a learned function to compute the importance score of the  $i$ -th adapter concerning the input  $\mathbf{x}$ , which is then normalized by a softmax function.

The mixed speech expert system dynamically integrates the outputs of each adapter through a gating mechanism that allows the system to adaptively adjust the contribution of each adapter according to the current input features and contextual demands. This dynamic integration process can be represented as:

$$O(\mathbf{x}) = \sum_{i=1}^n \alpha_i A_i(\mathbf{x}), \quad (4)$$

where  $O(\mathbf{x})$  is the total output of the system,  $A_i(\mathbf{x})$  is the output of the  $i$ -th adapter, and  $\alpha_i$  is the dynamic weight determined by the gating function  $G(\mathbf{x})$ , which reflects the importance of the adapters under the current input.

### 3.2 Mixture of Memory

Traditional experience replay methods [15, 22] retain a portion of old task samples and combine them with new task data to form a training set. We store mixed speech samples containing semantic events and acoustic events in memory, named mixed memory. We store these mixed samples in a memory buffer and use them with new data during model training.

Mathematically, it is assumed that the memory buffer  $\mathcal{M}$  contains the selected set of samples  $(x_i, y_i)$ , where  $x_i$  is the sample feature and  $y_i$  is the corresponding label. In the training phase, the

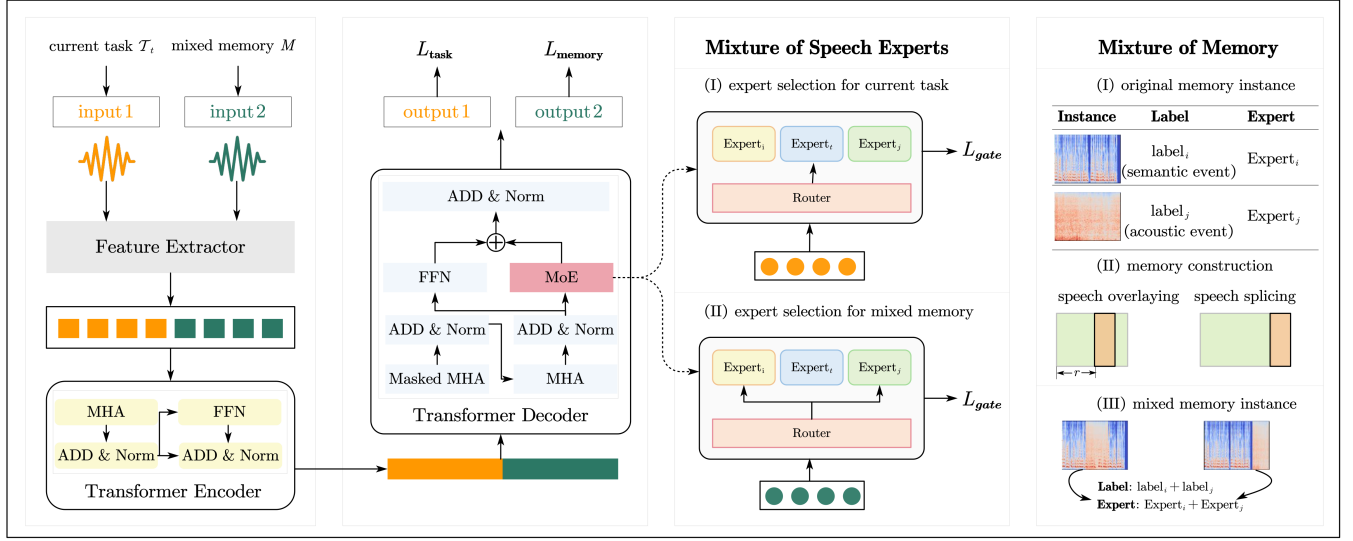


Figure 2: Framework of the proposed Double Mixture method.

model learns while drawing samples from the memory buffer and the current task data. The data-centric loss function is:

$$\mathcal{L}_{data} = \lambda \mathcal{L}_{task} + (1 - \lambda) \mathcal{L}_{memory}, \quad (5)$$

where  $\mathcal{L}_{task}$  and  $\mathcal{L}_{memory}$  are the cross-entropy loss, measuring the difference between the predicted probability distribution over the vocabulary and the actual distribution (one-hot encoded target) for instances from the current task data and the mixed memory respectively.  $\lambda$  is a parameter used to balance the importance of old and new knowledge, and the value is 0.5 based on experience.

### 3.3 Training Strategy

During the training phase, we freeze the parameters of the speech encoder to preserve its ability to understand speech features and only fine-tune the parameters of the mixture of speech experts in the decoding layer.

We compute the loss of the gating network to optimize the gating mechanism assigned by experts, and the loss function is:

$$\mathcal{L}_{gate} = - \sum_j |D| t_j \cdot \log \sum_{i=0}^d r_i, \quad (6)$$

where  $t_j$  is the gold task id of the  $j$ -th instance,  $d$  denotes the total number of decoder blocks, and  $r_i$  denotes the logits generated by the router.

Consequently, the aggregate training loss is formulated as a combination of the data-centric loss and the gating loss. The total loss is:

$$\mathcal{L}_{total} = \mathcal{L}_{data} + \eta \mathcal{L}_{gate}, \quad (7)$$

where  $\mathcal{L}_{data}$  focuses on the model's performance on the current training data,  $\mathcal{L}_{gate}$  ensures that the gating mechanism effectively integrates the outputs of multiple speech experts, and  $\eta$  is hyper-parameters used to balance out these two components of the loss.

## 4 EXPERIMENT

### 4.1 Datasets

We conduct experiments on three benchmark datasets, including the semantic event datasets Speech-ACE05 and Speech-MAVEN [11], and the acoustic event dataset ESC-50 [20]. We divide these datasets into a series of task sequences to satisfy continuous learning scenarios. Each task sequence corresponds to an independent speech event detection task to simulate the process of the model gradually being exposed to new tasks. The detailed data statistics are shown in Table 1.

Dataset	# Class	# Task	# Instance	# Hours
Speech ACE05	20	7	13984	33.36
Speech MAVEN	26	6	33666	81.28
ESC-50	50	5	2000	2.78

**Table 1: Dataset statistics. The event types in the dataset are two-level, and we assign task categories according to the first-level labels. For example, "Life" includes "Be-born", "Die", "Injure" and "Marry".**

To evaluate the generalization ability of the model in complex scenarios, we select the categories with a large number of samples from the above three speech data sets and then mix semantic events and acoustic events. Finally, we construct two new data sets, namely speech splicing (SS) and speech overlaying (SO). Table 2 shows their details.

### 4.2 Baselines

We compare our proposed model with the following baselines: (1) Fine-tuning (FT): We finetuned the Whisper model on several tasks sequentially. (2) Multi-task Learning (MTL): We finetuned the Whisper model on several tasks jointly. (3) Experience Replay

Task	Semantic	Sound	# Instance	SS (Avg sec.)	SS (Avg sec)
1	conflict	nature	2169	9.18	6.16
2	movement	nature	1057	4.47	3.01
3	scenario	animal	6955	29.42	19.72
4	talk	human	4695	19.86	13.34

**Table 2: Statistics information of the two combined datasets, SS and SO are the abbreviations of Speech splicing and Speech overlay respectively, Avg sec. indicates the average duration seconds of each audio file.**

(ER) [22]: a rehearsal-based method that leverages a data buffer to store and replay selected experiences from previous tasks. (4) Averaged Gradient Episodic Memory (A-GEM) [3]: a rehearsal-based method that extends the concept of ER by enforcing inequality constraints on the loss function for past experiences. (5) Elastic Weight Consolidation (EWC) [12] a regularization-based method that adds regularization on parameters according to their importance to old tasks. (6) Learning without Forgetting (LwF) [13]: a regularization-based method that utilizes a dual-model architecture where a frozen copy of the model acts as a teacher for the current, active model (student). This configuration allows the student model to learn from both the new task and the teacher, enriching its learning trajectory without forsaking prior knowledge. (7) Piggyback (PB) [18]: an architecture-based method introduces an innovative architectural adjustment that involves the application of binary masks derived from learnable weights onto the frozen parameters of a base model. (8) Learning to Prompt (L2P) [31] employs a collection of learnable vectors that are dynamically integrated into a pre-trained model.

### 4.3 Metrics

We calculate the average accuracy and average forgetting rate of each method over the entire task sequence.

**Average Accuracy.** To assess the overall performance across all learned tasks, we calculate incrementally after each newly introduced task:

$$Avg\ ACC = \frac{1}{T} \sum_{i=1}^T R_{T,i}, \quad (8)$$

where  $R_{T,i}$  represents the performance metric on the  $i$ -th task after training on the  $T$ -th task.

**Forgetting Rate.** We quantify the average loss in performance on earlier tasks, denoted as Forgetting Rate:

$$Avg\ Forgetting = \frac{1}{T} \sum_{i=1}^T R_{i,i} - R_{T,i}, \quad (9)$$

where *Forgetting* spans the range  $(-\infty, +\infty)$ , with positive values indicating forgetting on the prior tasks, and negative ones suggesting performance improvement.

### 4.4 Implementation Details

We train the models for 10 epochs per task using AdamW [16] with an initial learning rate of 0.0001, and we use a batch size is 16. After each epoch, the learning rate is reduced by 20% if no validation performance improvement is observed. We clip the gradient L2 norm to 5 to enhance stability. We also freeze Whisper’s encoder and

enable automatic mixed-precision to reduce memory consumption and speed up training. We use whisper-base<sup>1</sup> as the backbone and apply greedy decoding at inference. We retain 10% of the data from each previous task, about 20 to 100 samples. The combination of speech in memory is different from the test set. The gating loss coefficient is set to 0.1.

## 5 RESULTS AND ANALYSIS

### 5.1 Main Results

**5.1.1 Comparison of Continual Learning Methods.** Table 3 reports the average ACC and average Forgetting metrics at the end of the curriculum for different methods, and we observe that: (1) our method achieves the best performance, with the proposed method achieving 51.72% accuracy on the Speech ACE05, compared to 45.89% for the replay-based ER method and only 30.97% for the regularization-based EWC method. This result emphasizes the advantages of the proposed method in balancing new knowledge learning with old knowledge retention. (2) The advantages shown by the replay-based methods (ER and A-GEM) may stem from their ability to directly utilize historical data for retraining, where ER achieves 45.89% on the Speech ACE05 dataset, which is significantly higher than that of the EWC (30.97%) and LwF (24.56%). This reveals the effectiveness of the replay mechanism in promoting model memory retention. (3) Regularization-based methods perform poorly in the experiments, e.g., the accuracy of EWC is only 30.97%. This may be because while regularization constraints can help the model retain old knowledge, they may also limit the model’s learning of new tasks, especially when faced with new tasks that differ significantly from previous tasks. (4) Dynamic architecture methods (e.g., PB and L2P) provide the ability to dynamically adjust the model structure but still fall short in terms of forgetting rate, e.g., the forgetting rate of PB on Speech ACE05 is 59.89%, which is higher than that of the proposed method, which is 34.33%. This implies that although the dynamic architecture can introduce new network structures for new tasks, there are still challenges in long-term memory retention.

**5.1.2 Comparison of Backbone Model.** We compare the performance of some baseline methods using Whisper and WavLM as backbone models. Note that WavLM is an Encoder-only architecture, and we concatenate an LSTM as a text decoder after its speech Encoder. As shown in Table 5, we observe that Whisper performs better than WavLM in all three methods. Moreover, WavLM cannot identify acoustic events in the ESC-50 dataset. Therefore, we use Whisper in subsequent experiments.

**5.1.3 Exploration of Event Disentanglement.** We further evaluate the event disentanglement ability of the proposed method, and the experimental results are summarized in Table 4. We find that (1) On the SS dataset, the proposed method performs better in terms of average accuracy compared to the FT method and the ablation variants and similarly shows the same trend on the SO dataset. (2) The ablation experiments show that removing either the expert group or the memory mechanism leads to performance degradation, which confirms the importance of these two components in our method. (3) From the results, it can be observed that the average accuracy of Speech splicing is generally higher than that of Speech overlaying,

<sup>1</sup><https://huggingface.co/openai/whisper-base>

Method	Speech ACE05		Speech MAVEN		ESC-50	
	Avg ACC↑	Avg Forgetting↓	Avg ACC↑	Avg Forgetting↓	Avg ACC↑	Avg Forgetting↓
FT	31.10	76.59	31.28	91.24	38.98	85.62
ER	45.89	37.42	62.44	34.77	64.96	22.29
A-GEM	38.23	76.59	53.36	63.52	57.34	60.34
EWC	30.97	85.43	37.30	85.20	42.34	80.81
LwF	24.56	69.60	45.68	68.18	46.97	54.22
PB	21.15	59.89	37.36	78.67	37.73	66.58
L2P	42.91	61.60	53.15	46.46	51.97	38.99
<b>Double Mixture</b>	<b>51.72</b>	<b>34.33</b>	<b>65.53</b>	<b>24.32</b>	<b>73.12</b>	<b>21.30</b>
MTL	60.1	-	66.1	-	84.5	-

**Table 3: Main results on three benchmarks. All the results come from our implemented models. The numbers in the table are the average results at the end of a task flow. For each dataset, we mark the best representation in bold. Our method consistently outperforms other continual learning baselines on the three datasets.**

Speech splicing													
Task	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	Avg ACC↑	Avg Forgetting↓
FT	86.00	44.75	32.53	19.90	22.34	16.4	20.31	17.78	15.8	14.22	12.93	27.63	68.30
<b>Double Mixture</b>	87.00	82.75	49.73	22.90	15.74	18.13	14.55	21.77	24.91	30.71	35.04	<b>36.57</b>	58.89
w/o Experts	86.00	77.00	45.80	20.70	17.26	15.78	14.71	20.65	22.25	27.27	30.96	34.40	66.13
w/o Memory	86.00	81.75	40.20	21.12	17.93	13.73	14.37	20.85	21.5	26.74	29.98	34.02	66.47

Speech overlay													
Task	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	Avg ACC↑	Avg Forgetting↓
FT	83.00	44.75	32.53	19.90	22.34	16.4	20.31	17.78	15.8	14.22	12.93	27.63	71.01
<b>Double Mixture</b>	87.00	62.50	43.60	20.46	16.00	15.57	14.40	19.04	20.13	25.48	29.05	<b>31.75</b>	61.09
w/o Experts	85.00	47.25	27.60	21.01	16.76	15.86	14.45	18.04	17.16	19.93	21.55	27.39	68.71
w/o Memory	85.00	47.00	41.00	19.91	16.76	15.97	13.44	16.19	15.9	18.69	20.38	28.20	68.30

**Table 4: Performance on incremental tasks in Speech splicing and Speech overlay datasets. T1-T7 are tasks in which single semantic and acoustic events are randomly ordered, and T8-T11 are tasks in which two types of events are combined in speech. The latter four tasks are used to evaluate the generalization ability of the model, and their instances are not included in the training set. Given that other continual learning baseline methods can only predict single events and cannot identify combined events, they are not recorded in the table.**

Backbone	Whisper-base	WavLM-base
FT	31.10	21.03
ER	45.89	28.3
EWC	30.97	20.72
Double Mixture	51.72	-
MTL	60.1	39.6

**Table 5: Average accuracy (%) on Speech ACE05. Note that WavLM, being an encoder-only architecture, cannot implement our method as it requires a decoder.**

which may be because the superimposed speech samples have more interference in the acoustic signal, which increases the difficulty of the model in recognizing and distinguishing different events.

In addition, the interweaving of acoustic elements in the overlay samples may lead to more information loss, whereas the spliced examples maintain more information about the context of the event, providing a clearer basis for the model to make judgments.

## 5.2 Ablation Studies

**5.2.1 Importance of Each Component.** As can be seen in Figure 3, the accuracy of model recognition events decreases when speech experts are removed; this leads to a loss of about 21% in Speech ACE05, 25% in Speech MAVEN, and 32% in ESC-50. Similar findings are seen throughout the examined datasets, indicating that the model’s performance is significantly harmed by memory reduction.

**5.2.2 Impact of Task Order.** We investigate the impact of task ordering with our method on the ESC-50 datasets (See Table 6 and 7) and Speech MAVEN (See Table 8 and 9). Results show that different task orders resulted in differences in overall accuracy. For example, sequence #2 has an average accuracy of 73.06%, but sequence #4 yields an accuracy of 69% (in ESC-50). This demonstrates that even small adjustments to task order can affect model generalization and the ability to apply learned knowledge. However, our method still outperforms other baseline methods at the worst performance level. Similar observations can be found in the Speech MAVEN dataset.

Order	Task Sequence
#1	<i>animal → domestic → human → nature → urban</i>
#2	<i>domestic → nature → human → urban → animal</i>
#3	<i>domestic → urban → nature → animal → human</i>
#4	<i>animal → urban → nature → human → domestic</i>
#5	<i>domestic → urban → animal → human → nature</i>

Table 6: Random different task sequences on ESC-50.

Order	T1	T2	T3	T4	T5	Avg
#1	81.00	76.85	67.87	69.38	66.52	72.32
#2	85.00	70.00	72.10	69.40	68.82	73.06
#3	85.00	71.25	62.00	69.47	67.50	71.04
#4	81.00	72.60	68.27	64.12	59.02	69.00
#5	85.00	71.25	62.90	64.93	64.94	69.80

Table 7: Average accuracy (%) on incremental tasks in ESC-50.

Order	Task Sequence
#1	<i>process → action → scenario → move → life → talk</i>
#2	<i>action → process → move → life → scenario → talk</i>
#3	<i>scenario → process → action → move → life → talk</i>
#4	<i>move → talk → action → process → life → scenario</i>

Table 8: Random different task sequences on Speech MAVEN.

Order	T1	T2	T3	T4	T5	T6	Avg
#1	86.00	74.40	64.00	57.10	52.46	51.67	64.27
#2	76.00	72.35	61.90	56.50	54.52	53.23	62.42
#3	82.00	69.20	65.10	55.78	53.18	52.87	63.02
#4	79.00	70.35	62.33	61.65	55.34	50.97	63.27

Table 9: Average accuracy (%) on incremental tasks in Speech MAVEN.

## 6 RELATED WORK

### 6.1 Event Detection

Event Detection has been studied mostly focused on textual data, involving feature-based models [37] and deep learning models [30, 34, 39]. Recent work investigates the disparities between extracting events from text and speech, introducing an end-to-end approach for speech event extraction [11]. Conventional acoustic event detection methods [1, 19, 36] are typically characterized by a

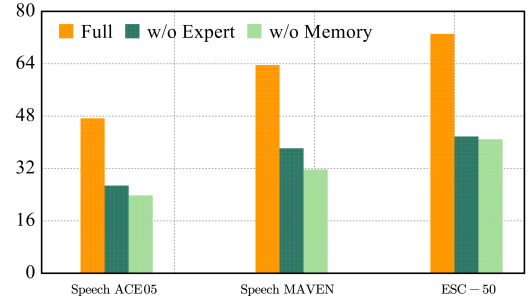


Figure 3: Ablation study on three datasets, the horizontal axis represents different data sets, and the vertical axis represents the average accuracy of the entire task sequence.

three-fold approach: (1) leveraging convolutional neural networks to extract acoustic features either from spectrogram representations or directly from audio waveforms; (2) employing recurrent neural networks or Transformers alongside a mean or max pooling mechanism to delineate temporal relationships among frame-level features; and (3) calculating the Connectionist Temporal Classification objective function.

### 6.2 Continual Learning

Existing continual learning algorithms [29, 33, 42] can be categorized into three main groups: (1) *Rehearsal-based methods* [3, 15, 22] involve either storing a small subset of training samples from previous tasks in memory or employing a data generator to produce pseudo samples of past tasks. When learning a new task, both retained and generated samples, along with new task data, are utilized for training. (2) *Regularization-based methods* [12, 13] apply penalties or regularization to discourage changes to important parameters acquired from previous tasks when learning a new task. (3) *Architecture-based methods* [18, 23, 31] allocate distinct parameters to different tasks to prevent interference with previously learned parameters, necessitating task identification during both training and testing. While these techniques mainly originate from computer vision [5] and are widely used in natural language processing [14, 32, 41], there is a noticeable research gap on continual speech learning, with that limited studies [8, 24, 26, 38] have explored continual learning for monolingual and multilingual ASR tasks.

## 7 CONCLUSION

This paper introduces a new task designed to continually extract both semantic and acoustic events from speech, focusing on overcoming catastrophic forgetting and improving event disentanglement. We propose a novel approach called "Double Mixture," employing a combination of Mixture of Experts and Mixture of Memory mechanisms. This strategy effectively mitigates catastrophic forgetting and enhances the model's ability to generalize across different types of events, outperforming existing continual learning benchmarks in terms of accuracy and forgetting metrics. Future work will explore applying this approach to broader speech processing and understanding tasks.



## REFERENCES

- [1] Çağdaş Bilen, Giacomo Ferroni, Francesco Tuveri, Juan Azcarreta, and Sacha Krstulović. 2020. A framework for the robust evaluation of sound event detection. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 61–65.
- [2] Heng-Jui Chang, Hung-yi Lee, and Lin-Shan Lee. 2021. Towards Lifelong Learning of End-to-End ASR. In *Interspeech 2021, 22nd Annual Conference of the International Speech Communication Association, Brno, Czechia, 30 August - 3 September 2021*. 2551–2555.
- [3] Arslan Chaudhry, Marc’Aurelio Ranzato, Marcus Rohrbach, and Mohamed El-hoseiny. 2019. Efficient Lifelong Learning with A-GEM. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6–9, 2019*. OpenReview.net. [https://openreview.net/forum?id=HkF2\\_sC5FX](https://openreview.net/forum?id=HkF2_sC5FX)
- [4] Boli Chen, Guangwei Xu, Xiaobin Wang, Pengjun Xie, Meishan Zhang, and Fei Huang. 2022. Aishell-ner: Named entity recognition from chinese speech. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 8352–8356.
- [5] Xiang Chen, Jintian Zhang, Xiaohan Wang, Tongtong Wu, Shumin Deng, Yongheng Wang, Luo Si, Huaun Chen, and Ningyu Zhang. 2023. Continual Multimodal Knowledge Graph Construction. *CoRR* abs/2305.08698 (2023). <https://doi.org/10.48550/arXiv.2305.08698>
- [6] Zitian Chen, Yikang Shen, Mingyu Ding, Zhenfang Chen, Hengshuang Zhao, Erik G Learned-Miller, and Chuang Gan. 2023. Mod-Squad: Designing Mixtures of Experts As Modular Multi-Task Learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 11828–11837.
- [7] Yashar Deldjoo, Markus Schedl, Paolo Cremonesi, and Gabriella Pasi. 2020. Recommender systems leveraging multimedia content. *ACM Computing Surveys (CSUR)* 53, 5 (2020), 1–38.
- [8] Luca Della Libera, Pooneh Mousavi, Salah Zaiem, Cem Subakan, and Mirco Ravanelli. 2023. CL-MASR: A Continual Learning Benchmark for Multilingual ASR. *arXiv preprint arXiv:2310.16931* (2023).
- [9] William Fedus, Barret Zoph, and Noam Shazeer. 2022. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *The Journal of Machine Learning Research* 23, 1 (2022), 5232–5270.
- [10] Yuan Gong, Sameer Khurana, Leonid Karlinsky, and James Glass. 2023. Whisper-at: Noise-robust automatic speech recognizers are also strong general audio event taggers. *arXiv preprint arXiv:2307.03183* (2023).
- [11] Jingqi Kang, Tongtong Wu, Jinming Zhao, Guitao Wang, Guilin Qi, Yuan-Fang Li, and Gholamreza Haffari. 2024. Towards Event Extraction from Speech with Contextual Clues. *CoRR* abs/2401.15385 (2024). <https://doi.org/10.48550/arXiv.2401.15385>
- [12] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. 2017. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences* 114, 13 (2017), 3521–3526.
- [13] Zhizhong Li and Derek Hoiem. 2017. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence* 40, 12 (2017), 2935–2947.
- [14] Minqian Liu, Shiyu Chang, and Lifu Huang. 2022. Incremental Prompting: Episodic Memory Prompt for Lifelong Event Detection. In *Proceedings of the 29th International Conference on Computational Linguistics, COLING 2022, Gyeongju, Republic of Korea, October 12–17, 2022*. 2157–2165.
- [15] David Lopez-Paz and Marc’Aurelio Ranzato. 2017. Gradient episodic memory for continual learning. *Advances in neural information processing systems* 30 (2017).
- [16] Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101* (2017).
- [17] Hui Lu, Xixin Wu, Zhiyong Wu, and Helen Meng. 2023. SpeechTripleNet: End-to-End Disentangled Speech Representation Learning for Content, Timbre and Prosody. In *Proceedings of the 31st ACM International Conference on Multimedia, MM 2023, Ottawa, ON, Canada, 29 October 2023 - 3 November 2023*. 2829–2837.
- [18] Arun Mallya, Dillon Davis, and Svetlana Lazebnik. 2018. Piggyback: Adapting a single network to multiple tasks by learning to mask weights. In *Proceedings of the European conference on computer vision (ECCV)*. 67–82.
- [19] Irene Martín-Morató, Manu Harju, Paul Ahokas, and Annamaria Mesáros. 2023. Training sound event detection with soft labels from crowdsourced annotations. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 1–5.
- [20] Karol J Piczak. 2015. ESC: Dataset for environmental sound classification. In *Proceedings of the 23rd ACM international conference on Multimedia*. 1015–1018.
- [21] Carlos Riquelme, Joan Puigcerver, Basil Mustafa, Maxim Neumann, Rodolphe Jenatton, André Susano Pinto, Daniel Keysers, and Neil Houlsby. 2021. Scaling vision with sparse mixture of experts. *Advances in Neural Information Processing Systems* 34 (2021), 8583–8595.
- [22] David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy Lillicrap, and Gregory Wayne. 2019. Experience replay for continual learning. *Advances in Neural Information Processing Systems* 32 (2019).
- [23] Joan Serra, Didac Suris, Marius Miron, and Alexandros Karatzoglou. 2018. Overcoming catastrophic forgetting with hard attention to the task. In *International conference on machine learning*. PMLR, 4548–4557.
- [24] Meysam Shamsi, Anthony Larcher, Loïc Barrault, Sylvain Meignier, Yevhenii Prokopal, Marie Tahon, Ambuj Mehrish, Simon Petitrenaud, Olivier Galibert, Samuel Gaist, André Anjos, Sébastien Marcel, and Marta R. Costa-jussà. 2023. Towards lifelong human assisted speaker diarization. *Comput. Speech Lang.* 77 (2023), 101437.
- [25] Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. 2017. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538* (2017).
- [26] Steven Vander Eeck and Hugo Van Hamme. 2022. Continual learning for monolingual end-to-end automatic speech recognition. In *2022 30th European Signal Processing Conference (EUSIPCO)*. IEEE, 459–463.
- [27] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).
- [28] Gary Wang, Kyle Kastner, Ankur Bapna, Zhehuai Chen, Andrew Rosenberg, Bhuvana Ramabhadran, and Yu Zhang. 2023. Understanding shared speech-text representations. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 1–5.
- [29] Liyan Wang, Xingxing Zhang, Hang Su, and Jun Zhu. 2023. A Comprehensive Survey of Continual Learning: Theory, Method and Application. *CoRR* abs/2302.00487 (2023).
- [30] Zitao Wang, Xinyi Wang, and Wei Hu. 2023. Continual Event Extraction with Semantic Confusion Rectification. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. 11945–11955.
- [31] Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. 2022. Learning to prompt for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 139–149.
- [32] Tongtong Wu, Massimo Caccia, Zhuang Li, Yuan-Fang Li, Guilin Qi, and Gholamreza Haffari. 2022. Pretrained Language Model in Continual Learning: A Comparative Study. In *The Tenth International Conference on Learning Representations*. <https://openreview.net/forum?id=figzpGMrdD>
- [33] Tongtong Wu, Linhao Luo, Yuan-Fang Li, Shirui Pan, Thuy-Trang Vu, and Gholamreza Haffari. 2024. Continual Learning for Large Language Models: A Survey. *CoRR* abs/2402.01364 (2024). <https://doi.org/10.48550/arXiv.2402.01364>
- [34] Tongtong Wu, Fatemeh Shiri, Jingqi Kang, Guilin Qi, Gholamreza Haffari, and Yuan-Fang Li. 2023. KC-GEE: knowledge-based conditioning for generative event extraction. *World Wide Web (WWW)* 26, 6 (2023), 3983–3999. <https://doi.org/10.1007/s11280-023-01216-5>
- [35] Tongtong Wu, Guitao Wang, Jinming Zhao, Zhaoran Liu, Guilin Qi, Yuan-Fang Li, and Gholamreza Haffari. 2022. Towards relation extraction from speech. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7–11, 2022*. 10751–10762. <https://doi.org/10.18653/v1/2022.emnlp-main.738>
- [36] Liang Xu, Lihong Wang, Sijun Bi, Hanyue Liu, and Jing Wang. 2023. Semi-Supervised Sound Event Detection with Pre-Trained Model. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 1–5.
- [37] Bishan Yang and Tom Mitchell. 2016. Joint Extraction of Events and Entities within a Document Context. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 289–299.
- [38] Mu Yang, Shaojin Ding, Tianlong Chen, Tong Wang, and Zhangyang Wang. 2022. Towards Lifelong Learning of Multilingual Text-to-Speech Synthesis. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2022, Virtual and Singapore, 23–27 May 2022*. 8022–8026.
- [39] Sen Yang, Dawei Feng, Linbo Qiao, Zhigang Kan, and Dongsheng Li. 2019. Exploring pre-trained language models for event extraction and generation. In *Proceedings of the 57th annual meeting of the association for computational linguistics*. 5284–5294.
- [40] Gang Yi, Dan Yang, Abdelhak Bentaleb, Weihua Li, Yi Li, Kai Zheng, Jiangchuan Liu, Wei Tsang Ooi, and Yong Cui. 2019. The ACM multimedia 2019 live video streaming grand challenge. In *Proceedings of the 27th ACM International Conference on Multimedia*. 2622–2626.
- [41] Pengfei Yu, Heng Ji, and Prem Natarajan. 2021. Lifelong Event Detection with Knowledge Transfer. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7–11 November, 2021*. 5278–5290.
- [42] Da-Wei Zhou, Hai-Long Sun, Jingyi Ning, Han-Jia Ye, and De-Chuan Zhan. 2024. Continual Learning with Pre-Trained Models: A Survey. *CoRR* abs/2401.16386 (2024).
- [43] Yanqi Zhou, Tao Lei, Hanxiao Liu, Nan Du, Yanping Huang, Vincent Zhao, Andrew M Dai, Quoc V Le, James Laudon, et al. 2022. Mixture-of-experts with expert choice routing. *Advances in Neural Information Processing Systems* 35 (2022), 7103–7114.