INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

Dual-Memory Multi-Modal Learning for Continual Spoken Keyword Spotting with Confidence Selection and Diversity Enhancement

Zhao Yang[1,3], *Dianwen Ng*[2,3], *Xizhe Li*[1], *Chong Zhang*[2], *Rui Jiang*[1], *Wei Xi*[1], *Yukun Ma*[2], *Chongjia Ni*[2], *Jizhong Zhao*[1], *Bin Ma*[2], *Eng Siong Chng*[3]

[1]Faculty of Electronic and Information Engineering, Xi'an Jiaotong University, China
[2]Speech Lab of DAMO Academy, Alibaba Group [3]Nanyang Technological University, Singapore

zhaoyang9425@gmail.com, dianwen.ng@alibaba-inc.com, {xiwei,zjz}@xjtu.edu.cn, aseschng@ntu.edu.sg

Abstract

Enabling continual learning (CL) from an ever-changing environment is highly valuable, but it poses significant challenges for spoken keyword spotting (KWS), which simultaneously deals with both variability in acoustic characteristics of speech signals and catastrophic forgetting issues. In this paper, we propose a novel framework for replay-based CL in KWS that uses a Dual-Memory Multi-Modal (**DM3**) structure to enhance generalizability and robustness. Our approach leverages short-term and long-term models to learn near-term and long-term knowledge in an adaptive manner with a dual-memory structure, while also exploiting the consistency of multiple speech perturbations to improve the robustness with a multi-modal structure. Additionally, we introduce a class-balanced selection strategy that uses confidence scores to sort training samples. Experiments demonstrate the effectiveness of our method over competitive baselines in class incremental learning and domain incremental learning KWS settings.

Index Terms: Keyword Spotting, Continual Learning, Confidence Selection, Diversity Enhancement

1. Introduction

Spoken keyword spotting (KWS) [1, 2, 3, 4] is the task of identifying specific keyword phrases within spoken utterances. It serves as a foundational component on many mobile and edge devices, such as voice assistants. Massive end-to-end models [3, 5, 6, 7, 8, 9] have demonstrated impressive performances on standard KWS tasks.

With the increasing interconnectedness of the world, the need for KWS systems that can adapt and learn on-the-fly has become more acute. Nevertheless, the approach of re-training a model from scratch for each new task entails an enormous amount of time and cost, which is often not an ideal solution. Continual learning (CL) for KWS presents several challenges due to several factors. One major challenge is developing a model that can adapt to dynamic environments where the data distribution is constantly changing while avoiding catastrophic forgetting problem [10]. Another challenge is the limited availability of annotated data and the non-stationary characteristics of the speech signals, making it difficult for the model to generalize effectively. Addressing these challenges requires careful consideration of the design of CL algorithms.

As far as we know, related works enabling CL for KWS tasks appear to be infrequently studied. [11] proposes a network instantiator to generate the task-specific sub-networks for remembering previously learned keywords. [12] introduces a replay-based method in which a diversity-aware sampler is designed to select a diverse set from historical and incoming keywords by calculating classification uncertainty. Other related work [13, 14, 15] involves few-shot fine-tuning [16] to perform positive transfer from source domain for new scenarios in KWS task. All these methods are mainly designed to preserve knowledge from previous tasks, without much emphasis on speech characteristics.

Given the effectiveness of the replay-based approach [12, 17] in CL tasks, we propose a novel replay-based CL approach that addresses the issues mentioned earlier. Our approach, called dual-memory multi-modal, aims to improve generalizability and robustness. To achieve this, we utilize a dual-memory structure that maintains both short-term and long-term semantic models. This is inspired by the brain working mechanism[18] and helps us to accumulate and consolidate information in a more effective manner. The short-term model performs well on recent tasks, while the long-term model prioritizes retaining information on older tasks. In addition, we exploit the consistency of speech diversity by multiple speech perturbations to enhance the model robustness with a multi-modal structure. This enhances the performance of our method in challenging environments where the speech signal may be corrupted or degraded. To mitigate catastrophic forgetting, we introduce a class-balanced selection strategy over sorted training samples based on confidence scores similar to curriculum learning. This helps to improve the overall performance of our method without significantly increasing training time. Our method is evaluated on a continual KWS benchmark, and experiments demonstrate its effectiveness and superiority over competitive baselines in both class incremental learning and domain incremental learning settings. To sum up, our contributions are threefold:

- A replay-based dual-memory multi-modal framework for continual KWS tasks is proposed, which consists of a dual-memory structure that better balances knowledge learning between long-term and short-term periods and a multi-modal structure that enhances robust representation with multiple diversity speech inputs.
- A class-balanced memory selection rule that selects diverse and representative replaying examples is proposed.
- Extensive experiments conducted on a continual KWS benchmark demonstrate the effectiveness of our proposed methods compared to competitive baselines.

2. Methodology

2.1. Problem Formulation

In this work, we focus on CL for a sequence of keyword spotting tasks $\{\tau_1, \tau_2, ..., \tau_T\}$, where we learn a model $f_\theta(.)$, θ is a set of parameters shared by all tasks. Each task τ_t contains a different set of training pairs (x_t, y_t) with distribution D^t, where x_t are the audio utterances and y_t are the keyword labels. We aim to

10.21437/Interspeech.2023-613

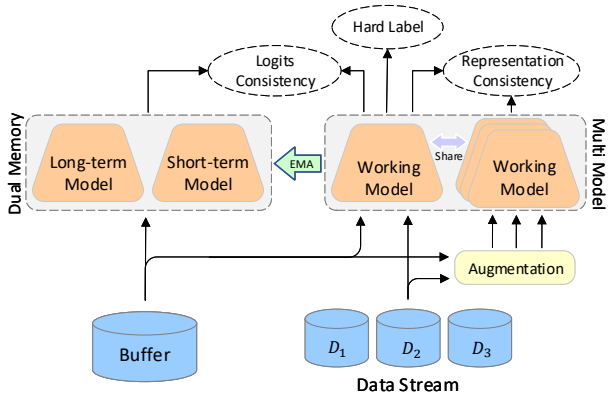

Figure 1: *The diagram of the proposed DM3 framework for continual KWS with confidence selection and diversity enhancement, which comprises of memory buffer, dual-memory component, and multi-modal component.*

minimize the generalization error on all tasks after learning all tasks in the sequence.

$$\mathcal{R}(f_\theta) = \sum_{t=0}^{T} \mathbb{E}_{(x_t,y_t)\sim D^t} \left[\mathcal{L}_{kws}(f(x_t;\theta), y_t)\right] \quad (1)$$

where $\mathcal{L}_{kws}$ denotes the cross-entropy loss.

In this work, we perform a comprehensive and thorough model evaluation under two CL settings: class incremental learning (**Class-IL**) and domain incremental learning (**Domain-IL**). In the Class-IL setting, different classes of data are added with each subsequent task and the model must learn to distinguish not only amongst the classes within the current task but also across previous tasks. In the Domain-IL setting, the classes remain the same in each subsequent task but the input distribution changes.

2.2. Dual-Memory Multi-Modal for Continual KWS

To deal with catastrophic forgetting and speech vulnerability problems, a **Dual-Memory Multi-Modal (DM3)** framework is proposed, which consists of three parts: memory buffer, dual-memory component and multi-modal component. The dual-memory and multi-modal components are formed with the same model unit. The detailed introduction of each module is as followed.

Memory Buffer In this work, the memory buffer is used as a fixed-size class-balanced buffer. Specifically, a fixed number of examples from each keyword class are stored in the memory buffer. This ensures that the model has access to a balanced representation of each class, even if the frequency of each class changes over time.

Model Unit TC-ResNet [7] utilizes the advantages of temporal convolution to enhance the accuracy and reduce the latency for real-time KWS on mobile devices. In our work, TC-ResNet-8 is chosen as the model unit of DM3 following [11, 12]. The model has three residual blocks and 16, 24, 32, 48 channels for each layer including the first convolution layer. The cross-entropy loss as one of the objective functions is used to update the model:

$$\mathcal{L}_{ce} = -\frac{1}{n}\sum_{i=1}^{n} y_i \, log(p_i) \quad (2)$$

where n is the keyword class.

Dual-Memory Structure The component maintains long-term and short-term models. The long-term model aims to accumulate and consolidate information throughout the training trajectory and the short-term model tends to be better at learning recent tasks. The logit of the one with the highest softmax score for the ground-truth class in both models is used as the final output (Line 8 in Algorithm 2). The long-term and short-term models are updated by taking an exponential moving average of the working model's weights θ_w:

$$\theta_{lt} \leftarrow \alpha_{lt}\theta_{lt} + (1-\alpha_{lt})\theta_w \quad if\ r_{lt} > a \sim \mathcal{U}(0,1) \quad (3)$$

$$\theta_{st} \leftarrow \alpha_{st}\theta_{st} + (1-\alpha_{st})\theta_w \quad if\ r_{st} > b \sim \mathcal{U}(0,1) \quad (4)$$

where θ_{lt} and θ_{st} are long-term and short-term models's weights respectively. α_{lt} and α_{st} are decay parameters. r_{lt} and r_{st} is the update rate. A higher update rate represents that the model is updated more frequently. The parameters r_{lt}, r_{st}, α_{lt} and α_{st} are set to 0.5, 0.9, 0.999 and 0.999 respectively.

We use an L2 loss based on the target representation from dual-memory output z and the working model prediction $\widehat{z}$:

$$\mathcal{L}_{mse} = |z - \widehat{z}|^2 \quad (5)$$

Multi-Modal Structure The component is designed to learn invariant representations under multiple distortions of a sample [19, 20]. Concretely, we first generate M different distorted views of training speech using a series of augmentations besides itself. Then the working model generates representations of one clean view and M distorted views. The component is trained by maximizing the similarity of representations between clean speech and distorted versions.

We use *Barlow Twins*'s objective function [21, 22] as a representation consistency metric donated by $\mathcal{L}_{cons}$. The overall objective function of DM3 for CL on KWS can be written as the following:

$$\mathcal{L}_{total} = \mathcal{L}_{ce} + \lambda\mathcal{L}_{mse} + \gamma\mathcal{L}_{cons} \quad (6)$$

where the weight parameter λ and γ are set to 0.15, 0.3 respectively.

2.3. Memory Buffer Selection Rule

Since only a small number of examples are stored in the memory buffer, we need to carefully select them in order to utilize the memory buffer $\mathcal{M}$ efficiently that balance diversity and representativeness [23]. The key to sample selection was to determine the learning priority of each sample similar to curriculum learning. The memory buffer selection rule is conducted after each task training (Line 20 Algorithm 2) and can be broken down into the following two steps:

Confidence Measurement The speech examples to be processed are from the current task, along with the memory buffer. By feeding each speech data into the working model, the softmax probability of the ground-truth class is obtained, which serves as the confidence score for the current sample. Then sorting the entire training set for each category in descending order of confidence scores.

Class-balance Memory Update Afterward, new speech examples in memory buffer are constructed by sampling at equal intervals in accordance with the class-balance strategy.

The memory selection algorithm is shown in Algorithm 1.

2.4. Training Formulation

During each training step, the main model receives the training batch X_b from current data stream $\mathcal{D}_t$ and a random batch of

Algorithm 1 Memory Buffer Selection Rule

Input: Memory buffer $\mathcal{M}$, Training data $\mathcal{D}_t$ for task t
Initialize: $\mathcal{D} = \mathcal{M} \bigcup \mathcal{D}_t$

for Sample $(x_i, y_i) \in \mathcal{D}$ **do**
  $z_i \leftarrow f(x_i; \theta_w)$
  $p_i \leftarrow \sigma(z_i)^{(y_i)}$ ▷ softmax score of ground-truth class
end for
for Each class k **do** ▷ select samples by class
  $\mathcal{D}^k_{sort} \leftarrow sort(\mathcal{D}^k; p)$
  $\mathcal{M}^k \leftarrow IntervalSampling(\mathcal{D}^k_{sort})$
end for
return $\mathcal{M}$

exemplars X_m from the memory buffer $\mathcal{M}$. The memory buffer is updated with Reservoir Sampling [24] in the initial task.

For inference, we use the working model since it learns efficient representations for generalization in our experimental setups. The training details are shown in Algorithm 2.

Algorithm 2 DM3 Learning for Continual KWS

Input: Training tasks $\{\tau_1, \tau_2, ..., \tau_T\}$, Update rates γ_p and γ_s, Perturbation strategies $\mathcal{P}_s$
Initialize: $\theta_w = \theta_{lt} = \theta_{st}$, $\mathcal{M} = \{\}$

for $t = 1, \ldots, T$ **do**
  for Batch $X_b \in \mathcal{D}_t$, $X_m \in \mathcal{M}$ **do**
    for $Aug \in \mathcal{P}_s$ **do**
      $X^a_b \leftarrow Aug(X_b)$
    end for
    $(X, Y) = \{(X_b, Y_b), *(X^a_b, Y_b), (X_m, Y_m)\}$
    $Z_{st}, Z_{lt} \leftarrow f(X_m; \theta_{st}), f(X_m; \theta_{lt})$
    $Z \leftarrow Z_{st}$ **if** $\sigma(Z_{st})^{(Y_m)} > \sigma(Z_{lt})^{(Y_m)}$ **else** Z_{lt}
    $\mathcal{L}_{ce} = CrossEntropy(X, Y)$
    $\mathcal{L}_{mse} = |Z - \sigma(f(X_m; \theta_w))|^2$
    $\mathcal{L}_{cons} = Consistency(f(X_b; \theta_w), f(*X^a_b; \theta_w))$
    Update multi-modal component with Eq. 6
    $a, b \sim \mathcal{U}(0, 1)$
    Update long-term model with Eq. 3
    Update short-term model with Eq. 4
    if $t = 1$ **then**
      $\mathcal{M} \leftarrow Reservoir(\mathcal{M}; (X_b, Y_b))$
    end if
  end for
  $\mathcal{M} \leftarrow MemorySelection(\mathcal{M} \bigcup \mathcal{D}_t)$
end for
return θ_w

3. Experiments

3.1. Dataset

we conduct experiments on the well-known *Google Speech Command V1* (GSC-V1) dataset [25]. It includes 64,727 one-second utterance clips with 30 English keyword categories. Following [11, 12], we first process all the data with a sample rate of 16kHz. We then split the dataset into two subsets, 80% for training and 20% for testing, respectively.

To verify and evaluate the performance of the proposed approach, raw data needs further processing to simulate CL scenarios. For the Class-IL setting, the first task contains 15 unique keywords and the rest data is split into 5 tasks. Each rest task includes 3 new unique keywords, which are unseen in previous tasks. For the Domain-IL setting, we split a subset of about 50% as the first task and the remaining 5 tasks contain 10% of the data. Each task contains all categories.

3.2. Experimental Setup

Training Details During the training stage, we utilize the Mel-frequency cepstrum coefficients (MFCC = 40) as inputs. The network is optimized by Adam with a learning rate of 0.1. All experiments are conducted on NVIDIA RTX 2080Ti with a batch size of 128 and training epochs of 50. Five perturbation strategies, including Clipping Distortion [26], TimeMask [26], Shift [27], PitchShift [27] and FrequencyMask [26] are applied as speech augmentation.

Compared Models *SGD* provides the lower bound with standard training on sequential tasks, and *JOINT* gives the upper bound on performance when the model is trained on the joint distribution. *EWC* [28] incorporates a quadratic penalty to regularize parameters of the model that are important to past tasks. *RWalk* [29] both calculate the importance for each parameter and store samples from past tasks. The others are all replay-based methods proposed in previous works.

Table 1: *Comparison with prior methods on Class-IL and Domain-IL settings.* $*$ *indicates that the values in this column need to be multiplied by 0.01.* ***Bold*** *values indicate the best result, underlined values indicate the second best result.*

Buffer	Method	Class-IL		Domain-IL	
		ACC ↑	BWT ↑	ACC ↑	BWT ↑ $*$
-	Joint	95.94	-	95.94	-
	SGD	36.19	-0.269	94.24	-0.247
-	EWC [28]	77.80	-0.088	94.14	-0.091
200	NR [30]	50.19	-0.189	94.32	-0.233
	iCaRL [31]	80.01	-0.084	91.92	-0.955
	BiC [32]	80.38	-0.080	91.91	-0.957
	RK [12]	81.46	-0.068	94.13	-0.253
	RWalk [29]	<u>84.33</u>	<u>-0.061</u>	**94.77**	**-0.020**
	DM3	**86.97**	**-0.038**	<u>94.56</u>	<u>-0.054</u>
500	NR [30]	57.64	-0.151	94.31	-0.223
	iCaRL [31]	85.35	-0.056	92.17	-0.999
	BiC [32]	83.08	-0.066	93.00	-0.407
	RK [12]	88.30	-0.041	93.89	-0.340
	RWalk [29]	<u>90.15</u>	<u>-0.032</u>	<u>94.76</u>	<u>-0.031</u>
	DM3	**91.77**	**-0.023**	**94.81**	**0.09**
1000	DM3	93.32	-0.016	94.91	0.063
2000	DM3	**93.80**	**-0.015**	**94.96**	**-0.032**

3.3. Experimental Results

We compare our method with regularization-based and replay-based methods across different CL settings in terms of Average Accuracy (ACC), Backward Transfer (BWT). As shown in Figure 1, the proposed method provides the highest performance for almost all buffer size conditions, which demonstrates the effectiveness of our approach in both Class-IL and Domain-IL settings. Furthermore, as the memory buffer size increases, the model performance will be further improved, especially in the Class-IL setting, which is closer to the upper bound performances.

Figure 2 shows how task-wise performance evolves as different models learn tasks sequentially. Compared to other models, our method consistently exhibits superior performance on the test set in the initial task (Column T1), which demonstrates

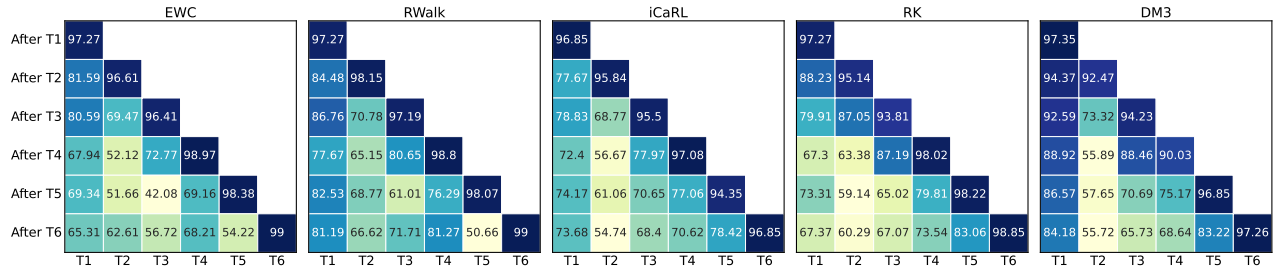

Figure 2: *The performance comparison of different task-wise methods with 200 buffer size. The heatmaps provide the test set of each task (x-axis) evaluated at the end of each sequential learning task (y-axis).*

the effectiveness of the long-term memory model. Meanwhile, our method achieves good performance on test sets of the current and near-term tasks in each sequential learning. However, we observe suboptimal performance in the mid-term task, especially as the number of tasks increases, which will be the focus of our future work.

3.4. Ablation Study

Impact of Memory Buffer Selection Rule To evaluate the effectiveness of the memory buffer selection rule, we analyze the performance differences with different selection rules. Table 2 illustrates the effectiveness of the class-balanced confidence sampling strategy. In the class-balanced setting, randomly sampled samples in the memory buffer with a small size may not effectively match the original data distribution. Moreover, without class-balanced constraints, samples of certain classes may be completely missing, which further exacerbates the degradation of model performance.

Table 2: *The performance comparison of different selection rules, i.e., class-balance, class-imbalance, interval sampling, random sampling, in terms of accuracy (%) on Class-IL and Domain-IL, respectively.*

Class-Balance	Sampling	Class-IL	Domain-IL
✓	Interval	**86.97**	**94.56**
✗	Interval	72.85	94.01
✓	Random	85.94	94.52
✗	Random	64.99	93.98

Impact of Model Architecture To gain further insight into the contribution of each component of our method, we systematically remove them and evaluate the performance of the model in Table 3. The results show that both dual-memory and multi-modal components contribute to the performance gains. Furthermore, the short-term and long-term models are likely complimentary to each other, and incorporating multi-view speech enhancement facilitates learning more robust representations.

Impact of Memory Update In order to study the effect of memory buffer update in the overall training process, we perform ablation study on the reservoir sampling and memory buffer selection rule. The results are summarized in Table 3. In conjunction with Figure 2, it is evident that utilizing the reservoir sampling strategy in initial task training has resulted in improved performance (97.35% *vs* 97.27%). After removing the memory buffer selection rule, the model training formulation is similar to [18], and the results show that a reasonable memory buffer update at the end of each task is necessary.

Impact of EMA update frequency Table 4 shows how the performance is affected under different frequencies for the long-term and short-term models. Although there are some variations, they are still higher than baseline models in Table 1, showing robustness of proposed method for update frequency.

Table 3: *Ablation study on model architecture. w/o augmentation models indicate removing the models fed with an augmented speech from the multi-modal structure. w/o MemorySelection indicates that the memory buffer selection rule is removed, while the reservoir sampling is applied across all tasks, not just the initial task.*

Model	Class-IL	Domain-IL
DM3 (200)	**86.97**	**94.56**
w/o Dual Memory	78.69	94.15
w/o Short-term	82.12	94.33
w/o Long-term	85.68	94.36
w/o Augmentation Models	86.09	94.40
w/ One Augmentation	86.76	94.34
w/o Reservoir	86.95	94.56
w/o MemorySelection	86.74	94.21

Table 4: *The performance comparison of the model at different update frequencies for long-term and short-term models.*

Long-term	Short-term	Class-IL	Domain-IL
0.2	0.9	86.81	94.35
0.3	0.9	86.60	94.36
0.5	0.9	**86.97**	94.56
0.6	0.7	86.53	**95.57**

4. Conclusion

Enabling CL in an ever-changing environment remains a challenge for KWS. In this paper, we presented a new CL framework called DM3 for KWS. Our empirical evaluation shows the effectiveness of the proposed approach in mitigating forgetting in challenging CL scenarios. Experimental results indicate the effectiveness of the proposed framework on standard KWS benchmarks including Class-IL and Domain-IL settings.

5. Acknowledgement

This work was supported in part by National Key R&D Program of China 2020YFB1707700, NSFC Grant No. 61832008, Alibaba Group through Alibaba Innovative Research (AIR) Program and China Scholarship Council.

6. References

[1] S. Ö. Arık, M. Kliegl, R. Child, J. Hestness, A. Gibiansky, C. Fougner, R. Prenger, and A. Coates, "Convolutional recurrent neural networks for small-footprint keyword spotting," *Proc. Interspeech 2017*, pp. 1606–1610, 2017.

[2] E. Yılmaz, O. B. Gevrek, J. Wu, Y. Chen, X. Meng, and H. Li, "Deep convolutional spiking neural networks for keyword spotting," in *Proceedings of INTERSPEECH*, 2020, pp. 2557–2561.

[3] D. Ng, Y. Chen, B. Tian, Q. Fu, and E. S. Chng, "Convmixer: Feature interactive convolution with curriculum learning for small footprint and noisy far-field keyword spotting," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 3603–3607.

[4] D. Ng, R. Zhang, J. Q. Yip, C. Zhang, Y. Ma, T. H. Nguyen, C. Ni, E. S. Chng, and B. Ma, "Contrastive speech mixup for low-resource keyword spotting," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[5] S. Majumdar and B. Ginsburg, "MatchboxNet: 1D Time-Channel Separable Convolutional Neural Network Architecture for Speech Commands Recognition," in *Proc. Interspeech 2020*, 2020, pp. 3356–3360.

[6] B. Kim, S. Chang, J. Lee, and D. Sung, "Broadcasted Residual Learning for Efficient Keyword Spotting," in *Proc. Interspeech 2021*, 2021, pp. 4538–4542.

[7] S. Choi, S. Seo, B. Shin, H. Byun, M. Kersner, B. Kim, D. Kim, and S. Ha, "Temporal convolution for real-time keyword spotting on mobile devices," *Proc. Interspeech 2019*, pp. 3372–3376, 2019.

[8] Y. Gong, Y.-A. Chung, and J. Glass, "AST: Audio Spectrogram Transformer," in *Proc. Interspeech 2021*, 2021, pp. 571–575.

[9] X. Li, X. Wei, and X. Qin, "Small-footprint keyword spotting with multi-scale temporal convolution," *Proc. Interspeech 2020*, pp. 1987–1991, 2020.

[10] M. McCloskey and N. J. Cohen, "Catastrophic interference in connectionist networks: The sequential learning problem," in *Psychology of learning and motivation*. Elsevier, 1989, vol. 24, pp. 109–165.

[11] Y. Huang, N. Hou, and N. F. Chen, "Progressive continual learning for spoken keyword spotting," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 7552–7556.

[12] Y. Xiao, N. Hou, and E. S. Chng, "Rainbow Keywords: Efficient Incremental Learning for Online Spoken Keyword Spotting," in *Proc. Interspeech 2022*, 2022, pp. 3764–3768.

[13] M. Mazumder, C. Banbury, J. Meyer, P. Warden, and V. J. Reddi, "Few-Shot Keyword Spotting in Any Language," in *Proc. Interspeech 2021*, 2021, pp. 4214–4218.

[14] A. Awasthi, K. Kilgour, and H. Rom, "Teaching Keyword Spotters to Spot New Keywords with Limited Examples," in *Proc. Interspeech 2021*, 2021, pp. 4254–4258.

[15] J. Lin, K. Kilgour, D. Roblek, and M. Sharifi, "Training keyword spotters with limited and synthesized speech data," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7474–7478.

[16] Q. Sun, Y. Liu, T.-S. Chua, and B. Schiele, "Meta-transfer learning for few-shot learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 403–412.

[17] F. Sarfraz, E. Arani, and B. Zonooz, "Sparse coding in a dual memory system for lifelong learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2023.

[18] E. Arani, F. Sarfraz, and B. Zonooz, "Learning fast, learning slow: A general continual learning method based on complementary learning system," in *International Conference on Learning Representations*, 2022. [Online]. Available: https://openreview.net/forum?id=uxxFrDwrE7Y

[19] A. Baevski, A. Babu, W.-N. Hsu, and M. Auli, "Efficient self-supervised learning with contextualized target representations for vision, speech and language," *arXiv preprint arXiv:2212.07525*, 2022.

[20] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. 2. IEEE, 2006, pp. 1735–1742.

[21] J. Zbontar, L. Jing, I. Misra, Y. LeCun, and S. Deny, "Barlow twins: Self-supervised learning via redundancy reduction," in *International Conference on Machine Learning*. PMLR, 2021, pp. 12 310–12 320.

[22] D. Ng, R. Zhang, J. Q. Yip, Z. Yang, J. Ni, C. Zhang, Y. Ma, C. Ni, E. S. Chng, and B. Ma, "De'hubert: Disentangling noise in a self-supervised model for robust speech recognition," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[23] Y. Huang, Y. Zhang, J. Chen, X. Wang, and D. Yang, "Continual learning for text classification with information disentanglement based regularization," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Online: Association for Computational Linguistics, Jun. 2021, pp. 2736–2746.

[24] J. S. Vitter, "Random sampling with a reservoir," *ACM Transactions on Mathematical Software (TOMS)*, vol. 11, no. 1, pp. 37–57, 1985.

[25] P. Warden, "Speech commands: A dataset for limited-vocabulary speech recognition," *arXiv preprint arXiv:1804.03209*, 2018.

[26] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "Specaugment: A simple data augmentation method for automatic speech recognition," *Proc. Interspeech 2019*, pp. 2613–2617, 2019.

[27] T. Ko, V. Peddinti, D. Povey, and S. Khudanpur, "Audio augmentation for speech recognition," in *Sixteenth annual conference of the international speech communication association*, 2015.

[28] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska *et al.*, "Overcoming catastrophic forgetting in neural networks," *Proceedings of the national academy of sciences*, vol. 114, no. 13, pp. 3521–3526, 2017.

[29] A. Chaudhry, P. K. Dokania, T. Ajanthan, and P. H. Torr, "Riemannian walk for incremental learning: Understanding forgetting and intransigence," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 532–547.

[30] Y.-C. Hsu, Y.-C. Liu, A. Ramasamy, and Z. Kira, "Re-evaluating continual learning scenarios: A categorization and case for strong baselines," in *NeurIPS Continual learning Workshop*, 2018.

[31] S.-A. Rebuffi, A. Kolesnikov, G. Sperl, and C. H. Lampert, "icarl: Incremental classifier and representation learning," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2017, pp. 2001–2010.

[32] Y. Wu, Y. Chen, L. Wang, Y. Ye, Z. Liu, Y. Guo, and Y. Fu, "Large scale incremental learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 374–382.