REDUX: AN ITERATIVE STRATEGY FOR SEMANTIC SOURCE SEPARATION

Technical Report

Vasileios Stergioulis *Gerasimos Potamianos*

Department of Electrical and Computer Engineering, University of Thessaly, Volos, Greece
vstergioulis@uth.gr gpotamianos@uth.gr

ABSTRACT

In this report, we present a system for the Spatial Semantic Segmentation of Sound Scenes (DCASE 2025 Task 4), combining enhanced source separation and label classification through an iterative verification strategy. Our approach integrates the Masked Modeling Duo (M2D) classifier with a separator architecture based on an attentive ResUNeXt network. Inspired by recent advances in universal audio modeling and self-supervised separation, our system incorporates feedback between multiple classification and separation stages to correct early-stage prediction errors. Specifically, classification outputs are verified using post-separation reclassification, and ambiguous cases are resolved through targeted waveform subtraction and re-analysis. This strategy enables improved source-label associations without increasing model complexity. Evaluated on the development set, our method achieves a relative improvement of 0.28% in CA-SDRi and 1.46% in accuracy over the baseline.

***Index Terms*—** Semantic Source Separation, Sound Classification, Source Separation, UNet, DCASE25

1. INTRODUCTION

The task of Spatial Semantic Segmentation of Sound Scenes (S5), introduced in the DCASE 2025 Challenge Task 4 [1], requires systems to simultaneously perform source separation and sound event classification on real-world multi-source mixtures. Unlike traditional sound event detection (SED) tasks, S5 demands not only temporal but also semantic disentanglement of individual sound sources from overlapping audio mixtures. Building on the baseline systems proposed by [2], which integrate the M2D [3] classifier with convolutional separators, we propose a modified framework that explicitly addresses the uncertainty in early predictions via an iterative consensus strategy.

Our method draws inspiration from recent work on self-supervised representation learning and universal source separation [4], [5], where intermediate feature verification is used to refine downstream tasks. Specifically, we enhance the separator network with an attentive ResUNeXt backbone, capable of spectral patterns via grouped convolutions and residual attention mechanisms. This separator is trained with a spectrogram masking objective and SpecAugment regularization to impove generalization.

What sets our system apart is the inclusion of an iterative classification-separation-classification loop. After an initial label estimation using M2D (Classifier 1), we perform source separation conditioned on these labels. The separated waveforms are then re-evaluated using a second classification pass (Classifier 2). Agreement between the two classification stages acts as a proxy for separation success; in cases of disagreement, our system applies selective waveform subtraction and reclassification. This strategy generalizes to any number of sources, enabling robust multi-source disentanglement.

2. DATASET

The development set provided for the S5 DCASE [6] Task 4 is comprised of sounds from FSD50K [7], EARS [8], FOA-MEIR [9], ESC-50 [10], DISCO [11] and some additional samples recorded at NTT labs. For the purposes of this year's task (and in our system) all the data are sampled at 32 kHz.

We also chose to employ SpecAugment [12] to the input spectrograms on our separator network, setting both the frequency mask parameter F and the time mask parameter T as 80 (default parameters). The augmented data are presented to our separator network, after training on the original development set data is performed. As shown in Section 4, by applying the masking operation, both CA-SDRi [2] and label prediction accuracy metrics improve.

3. SYSTEM DESCRIPTION

Our proposed system-method comprises different combinations of the same classifier and separator in an iterative consensus scheme as described in subsection 3.4. The system first classifies the scene, separates accordingly, and then uses agreement between two classification passes with the addition of targeted source subtraction to self-verify and repair separation errors. This strategy lets the system correct mis-separations without needing extra or different separators. The chosen classifier is M2D [3], i.e. the same one used in the baseline system [2]. In our work, we also implement a modified version of the baseline system separator with ResNeXt connections [13] and an attentive residual path [14], named ResUNeXt, as illustrated in Figure 1.

3.1. ResUNeXt

A ResNet block consists of a few layers that apply transformations and add the result back to the input (e.g., the residual connection). In our case, the ResNet block from [5], [15] consists of two convolutional layers with batch normalization and the residual path of a convolution. A ResNeXt block is an enhanced version, with a novel architectural parameter called "cardinality" (C), which is the number of parallel transformations within the block. This means that instead of one, multiple parallel paths (like a mini-ensemble) exist, performing the same transformations with their outputs being aggregated (summed, concatenated, or by grouped convolutions).

Cardinality is utilized as a new dimension of scaling, in a sense that scaling a neural network (e.g., increasing the accuracy) focuses

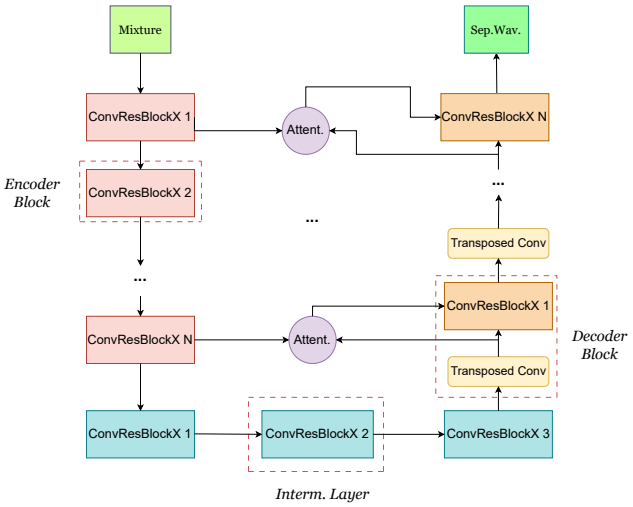

Figure 1: Attentive ResUNeXt (Figure modified from [14]).

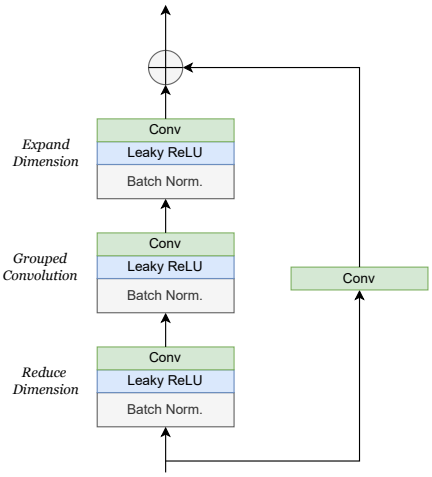

Figure 2: ResNeXt Block (Figure modified from [15]).

on two axes: increasing either the layers (depth) or the channels (width); Cardinality is introduced as a third axis that scales our neural network without increasing the layers or the width, and thus without increasing the computational cost. After a series of experiments, the cardinality value is set as equal to 8, while the size of the grouped convolutions (G_C) is chosen based on (1):

$$G_C = \lfloor \frac{4 \cdot O_{Ch}}{32} \rfloor \cdot C, \tag{1}$$

where O_{Ch} is the number of output channels in our convolutional block. Following the same logic as in [13], our convolutional blocks consist of three convolutions, preceded by batch normalization and the leaky ReLU activation function. Every convolution has a different usage: The first one is used to reduce the dimension based on the grouped convolution size taken from (1) and output G_C features-channels, the second one performs grouped convolutions based on cardinality producing G_C features-channels, while the last convolution expands our channels-features to the desired output dimension size. Note that every encoder and decoder block have different input and output feature sizes as shown in Table 1. Our separator network is trained using the SDR loss. The convolutional block is illustrated in Figure 2.

3.2. Residual Attentive Path

Another mechanism that is utilized in our separator is self-attention, as described in [14]. This module has the ability to preserve the key features of the target source while suppressing the features of the other components. It receives as input the output of the residual path of the corresponding level of the encoder and the transposed decoder output. The mechanism is a modified version of [16], with the addition of convolutional layers.

By fine-tuning the network's hyperparameters, the finalized version of the attentive path uses the leaky Rectified Linear Unit (leaky ReLU) as its activation function, as shown in Figure 3. E_{nc} and D_{ec} denote the encoder and decoder output, respectively.

3.3. Architecture Overview

The Attentive Multichannel ResUneXt takes as input the raw multichannel waveform and transforms it into the time-frequency domain using Short-Time Fourier Transform, following an encoder-decoder architecture. This spectrogram is then normalized and passed through a preliminary convolution to prepare the features for encoding. Each encoder block consists of eight ResNeXt blocks, incorporating grouped convolutions, and it outputs a downsampled representation and a high-resolution residual connection for the decoder. Following [15], the encoder consists of five encoder blocks and FiLM [17] layer conditioning to extract deep spectral features, while progressively downsampling the time-frequency resolution.

At the core of the network, three ResNeXt blocks operate at the most compressed resolution, enhancing the representational capacity of the ResUNeXt architecture. These are followed by a symmetric decoder that progressively upsamples the feature maps. Each decoder block is built in a similar manner as an encoder block consisting of eight ResNeXt blocks, while the entire decoder contains five decoder blocks. Note that before the decoder block input, a transposed convolution exists, so that the previous intermediate or decoder layer is properly used as input to the attentive residual path. This means that our encoder and decoder blocks use the same architecture, with the transposed convolution existing at the start of each decoder layer.

3.4. Strategy

Before we define our strategy-algorithm to address the S5 task [6], we need to reinstate the task problem. Let $\mathbf{Y} = [y^{(1)}, ...y^{(M)}]^{\mathcal{T}} \in \mathcal{R}^{M \times T}$ be the multichannel time-domain mixture signal of length T, recorded with an array of M microphones. Let also $\mathcal{C} = \{c_1, ..., c_K\}$ be the set of source labels in the mixture, where the source count K can vary from 1 to K_{max}. The $m-$th channel of $\mathbf{Y}$ can be modeled as:

$$\mathbf{y}^{(m)} = \sum_{k=1}^{K} \mathbf{h}_k^{(m)} * \mathbf{s}_k + \mathbf{n}^{(m)} = \sum_{k=1}^{K} \mathbf{x}_k^{(m)} + \mathbf{n}^{(m)}, \tag{2}$$

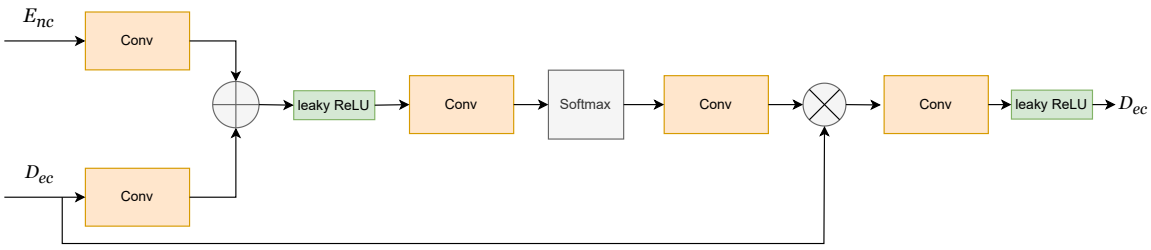

Figure 3: Attentive Residual path (Figure modified from [14]).

Encoder Blocks

Layer	Input Channels	Output Channels
Enc. Block 1	32	32
Enc. Block 2	32	64
Enc. Block 3	64	128
Enc. Block 4	128	256
Enc. Block 5	256	384

Intermediate Layers

Layer	Input Channels	Output Channels
Intermed. Block 1	384	384
Intermed. Block 2	384	384
Intermed. Block 3	384	384

Decoder Layers

Layer	Input Channels	Output Channels
Dec. Block 1	384	384
Dec. Block 2	384	256
Dec. Block 3	256	128
Dec. Block 4	128	64
Dec. Block 5	64	32

Table 1: Architecture of the Encoder, Intermediate, and Decoder blocks.

where $\mathbf{s}_k \in \mathcal{R}^T$ is our target monoaural channel source signal corresponding to label c_k, while $\mathbf{h}_k^{(m)}$ and $\mathbf{n}^{(\mathbf{m})}$ are the m-th channel room impulse response (RIR) and noise. The noisy-wet source $\mathbf{x}_k^{(m)}$ can be divided in two components: the direct path $\mathbf{h}_k^{(m,d)} * \mathbf{s}_k$ and the late reverberation $\mathbf{h}_k^{(m,r)} * \mathbf{s}_k$, where $\mathbf{h}_k^{(m,d)}$ and $\mathbf{h}_k^{(m,r)}$ are the corresponding early and late parts of the RIR, respectively.

In order to make our notation easier, let us assume that we have 3 sources, i.e. $K = 3$, and also ignore the noise (both reverbation and room noise) and let us also consider that we have only one microphone instead of an array of microphones (monoaural sound) to simplify our problem as shown in (3):

$$\mathbf{y} = \sum_{k=1}^{K} \mathbf{s}_k \overset{K=3}{\Rightarrow} \mathbf{y} = \mathbf{s}_1 + \mathbf{s}_2 + \mathbf{s}_3 \tag{3}$$

The main idea is to first identify the labels present in the input mixture using a classifier, denoted as C_1, and based on these predicted labels we can then generate the corresponding waveforms using our separator, denoted as Sep_1. After the separation we again use a classifier, denoted as C_2, which operates on the separated signals. Since identifying individual sources in isolated signals is significantly easier than doing so within a full mixture, C_2 acts as a validation step for the predictions made by C_1. If the separation is accurate (and therefore the classification), the second classifier should confidently and correctly label the sources. Note that symbols $C_1, C_2, Sep_1, Sep_2, ...$ refer to the sequence of classifiers and separators within the system pipeline, rather than to differences in model architecture. In practice, the same classifier and separator architectures may be reused across these stages. Our strategy is illustrated in Figure 4.

We start by passing our mixture through a classifier in order to obtain the labels and identify which sounds are present in the mixture to ease the separation process. Let $l_1, l_2, l_3, \bar{l}_1, \bar{l}_2, \bar{l}_3$ be the predicted labels of C_1 and C_2 respectively, and let $\mathbf{w}_1, \mathbf{w}_2, \mathbf{w}_3$ be the waveforms after the separation task. From here on, we have four possible cases: (I) all labels are the same, (II) two labels are the same, (III) one label is the same and (IV) no same labels.

For our purposes, we rewrite (3) as:

$$\mathbf{y} = \mathbf{w}_1 + \mathbf{w}_2 + \mathbf{w}_3 \tag{4}$$

3.4.1. *Case I*

For the first case where our two classifiers produced the same labels, we assume that both the label prediction and the separation task are successful and the process ends.

3.4.2. *Case II*

In the second case, different labels suggests that either one or both the classification and separation tasks have failed. Recalling (4), we can clearly understand that in order to find the third source we simply need to subtract from our mixture the estimated waveforms. Let us say that we identified and separated correctly the first and third sources, i.d. $l_1 = \bar{l}_1, l_3 = \bar{l}_3$ and $\mathbf{w}_1, \mathbf{w}_3$ are the estimated waveforms. The third estimation is then simply:

$$\mathbf{y} = \mathbf{w}_1 + \mathbf{w}_2 + \mathbf{w}_3 \Rightarrow \mathbf{w}_2 = \mathbf{y} - \mathbf{w}_1 - \mathbf{w}_3 \tag{5}$$

In order to find the remaining estimated waveform's label, we use a third classifier C_3 producing $\bar{\bar{l}}_2$ and ending the process. If $\bar{\bar{l}}_2$ is identical to one of the previous labels, then we consider $\mathbf{w}_2$ as silence and only produce two waveforms.

3.4.3. *Case III*

In the third case where only one label is correct, we subtract the corresponding waveform from our mixture, resulting in a new mixture $\mathbf{y}_1$. Let us assume that $l_1 = \bar{l}_1$ and $\mathbf{w}_1$ be the corresponding

waveform. Then the new mixture $\mathbf{y}_1$ is:

$$\mathbf{y}_1 = \mathbf{w}_2 + \mathbf{w}_3 \tag{6}$$

Here, we also take in note the probabilities of the previous classifiers in order to serve as a safeguard. We start by assigning to the remaining two waveforms the class labels based on the highest probabilities originating from C_1 and C_2. Now what is left to do is to again pass the new estimated waveform $\mathbf{y}_1$ through a classifier C_i, a separator Sep_i and a classifier C_{i+1}, in the same manner as before. The only change here is that we always choose the label associated with the highest probability originating from either the first two classifiers (C_1, C_2) or from the last two classifiers (C_i, C_{i+1}). In the case that silence is detected in our waveforms we assign it a probability value of 0.51, approximately the same as the probability threshold used in [2]. The process continues and we fall to one of the three remaining cases: (I), (II), or (IV).

3.4.4. Case IV

In the final case, where none of the labels are the same, we always choose the label with the highest probability, as predicted from the previous two classifiers. After choosing the labels, we use a second separator Sep_2 in order to receive the final waveforms.

3.5. Strategy Generalization

To generalize our strategy for K sources, we denote the mixture as $\mathbf{y} = \sum_{k=1}^{K} \mathbf{s}_k$, where $\mathbf{s}_k \in \mathcal{R}^T$ are the individual source signals corresponding to labels $c_k \in \mathcal{C}$. The system first employs a classifier C_1 to predict the set of active source labels $\{l_k\}_{k=1}^{K}$ along with their associated probabilities $\{p_{l_k}\}_{k=1}^{K}$. Based on these predictions, a separator Sep_1 generates the estimated waveforms $\{\mathbf{w}_k\}_{k=1}^{K}$, which are subsequently evaluated by a second classifier C_2, producing a second set of labels $\{\bar{l}_k\}_{k=1}^{K}$ and probabilities $\{\bar{p}_{l_k}\}_{k=1}^{K}$.

The agreement between the label sets $\{l_k\}$ and $\{\bar{l}_k\}$ guides the subsequent steps. For each pair $(l_k, \bar{l}_k)$, agreement implies reliable separation and classification, while disagreement indicates ambiguity. When fewer than K labels agree across classifiers, the remaining waveform estimates are refined through subtraction from the original mixture: for an unknown source j, $\mathbf{w}_j = \mathbf{y} - \sum_{k \neq j} \mathbf{w}_k$. These residuals are reclassified using new classifiers C_i until label agreement improves or silence is detected. This iterative process continues, branching into cases based on the degree of label overlap, until a consistent label-waveform mapping is achieved across all K estimated sources.

4. RESULTS

4.1. Development Set

In Table 2 we depict the comparison of the evaluation metrics [2] between the system baseline and our proposed system in the validation-test set (of the development set). We can readily observe that, compared to the baseline, we managed to achieve a 0.28% relative increase in the CA-SDRi score (from 11.088 to 11.119) and a 1.46% relative improvement in label prediction accuracy (from 59.8% to 60.67%).

The observed improvements in both CA-SDRi and classification accuracy stem directly from the design of our iterative classifier-separator strategy. Unlike the baseline systems, which perform classification and separation sequentially without verification, our method introduces a closed-loop mechanism where classification results are validated post-separation using a second classifier. This enables error correction and refined source-label associations. The strategy not only improves the quality of the source estimation, but also enhances the semantic understanding of the scene.

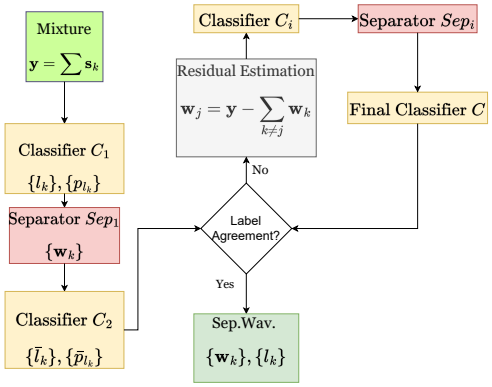

Figure 4: Strategy generalization.

System	CA-SDRi (dB)	Accuracy (%)
M2D + ResUNet (baseline)	11.032	59.80
M2D + ResUNetk (baseline)	11.088	59.80
M2D + Att. ResUNeXt w/o SpecAugment	11.035	59.80
M2D + Att. ResUNeXt	11.040	59.80
M2D + Att. ResUNeXt + strategy	**11.119**	**60.67**

Table 2: Results from the baseline and proposed systems on the validation-test set.

4.2. Evaluation Set

The results achieved from our submission in the evaluation set may be reported in a revised version of this technical report.

5. CONCLUSIONS AND FUTURE WORK

We presented a solution for Task 4 of the 2025 DCASE Challenge using a CNN based separator, addressing the separation and classification tasks as a whole and not separately. In particular we implemented a new type of separator using ResNeXt connections with the addition of an attentive residual path in order to better learn the spectrogram masks. By using our strategy we manage to validate our hypothesis that jointly optimizing classification and separation through iterative verification enhances both components of the system.

In future work, we will investigate possible improvements with the use of all microphones present in the microphone array with signal augmentation techniques or by applying weights for every channel prediction on the classification task. Finally, we will also explore a combination of different classifier and separator networks in our strategy in order to further improve our results.

6. REFERENCES

[1] M. Yasuda, B. T. Nguyen, N. Harada, R. Serizel, M. Mishra, M. Delcroix, S. Araki, D. Takeuchi, D. Niizumi, Y. Ohishi, T. Nakatani, T. Kawamura, and N. Ono, "Description and discussion on dcase 2025 challenge task 4: Spatial semantic segmentation of sound scenes," 2025, arXiv:2506.10676.

[2] B. T. Nguyen, M. Yasuda, D. Takeuchi, D. Niizumi, Y. Ohishi, and N. Harada, "Baseline systems and evaluation metrics for spatial semantic segmentation of sound scenes," https://arxiv.org/abs/2503.22088, 2025, arXiv:2503.22088.

[3] D. Niizumi, D. Takeuchi, Y. Ohishi, N. Harada, and K. Kashino, "Masked modeling duo: Towards a universal audio pre-training framework," *IEEE/ACM Trans. on Audio, Speech, and Lang. Process.*, 2024.

[4] E. Tzinis, S. Wisdom, J. R. Hershey, A. Jansen, and D. P. W. Ellis, "Improving universal sound separation using sound classification," in *IEEE Intl. Conf. on Acoust., Speech & Sig. Proc. (ICASSP)*, 2020, pp. 96–100.

[5] Q. Kong, K. Chen, H. Liu, X. Du, T. Berg-Kirkpatrick, S. Dubnov, and M. D. Plumbley, "Universal source separation with weakly labelled data," 2023, arXiv preprint arXiv:2305.07447.

[6] https://dcase.community/challenge2025/task-spatial-semantic-segmentation-of-sound-scenes/.

[7] E. Fonseca, X. Favory, J. Pons, F. Font, and X. Serra, "FSD50K: An open dataset of human-labeled sound events," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 829–852, 2022.

[8] J. Richter, Y.-C. Wu, S. Krenn, S. Welker, B. Lay, S. Watanabe, A. Richard, and T. Gerkmann, "EARS: An anechoic fullband speech dataset benchmarked for speech enhancement and dereverberation," in *Interspeech 2024*, 2024, pp. 4873–4877.

[9] M. Yasuda, Y. Ohishi, and S. Saito, "Echo-aware adaptation of sound event localization and detection in unknown environments," in *IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, 2022.

[10] K. J. Piczak, "ESC: Dataset for environmental sound classification," in *Proceedings of the 23rd ACM International Conference on Multimedia (MM '15)*. New York, NY, USA: Association for Computing Machinery, 2015, pp. 1015–1018.

[11] N. Furnon, "Noise files for the disco dataset," https://github.com/nfurnon/disco, 2020, accessed: 2025-06-08.

[12] 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," in *Interspeech*, 2019, pp. 2613–2617.

[13] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 5987–5995.

[14] T. Sgouros, A. Bousis, and N. Mitianoudis, "An efficient short-time discrete cosine transform and attentive multiresunet framework for music source separation," *IEEE Access*, vol. 10, pp. 119 448–119 459, 2022.

[15] Q. Kong, Y. Cao, H. Liu, K. Choi, and Y. Wang, "Decoupling magnitude and phase estimation with deep ResUNet for music source separation," in *International Society for Music Information Retrieval (ISMIR)*, 2021.

[16] O. Oktay, J. Schlemper, L. L. Folgoc, M. Lee, M. Heinrich, K. Misawa, K. Mori, S. McDonagh, N. Y. Hammerla, B. Kainz, B. Glocker, and D. Rueckert, "Attention U-Net: Learning where to look for the pancreas," 2018, arXiv:1804.03999.

[17] E. Perez, F. Strub, H. de Vries, V. Dumoulin, and A. Courville, "Film: Visual reasoning with a general conditioning layer," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.