Cosine similarity based Few-shot Bioacoustic Event Detection with Automatic Frequency Range Identification in Mel-Spectrograms

Technical Report

Sheng-Lun Kao[1], *Yi-Wen Liu*[1]

[1] Department of Electrical Engineering, National Tsing Hua University, Hsinchu, Taiwan
shenglun@gapp.nthu.edu.tw, ywliu@ee.nthu.edu.tw

ABSTRACT

In response to the Few-shot Bioacoustic Event Detection challenge, we have developed a detection system comprising three key components. First, an algorithm has been devised for automatically identifying frequency ranges of the positive (POS) signal within the mel-spectrogram. Secondly, the cosine similarity between POS and negative (NEG) events is computed across the entire audio file. Thirdly, predictions of POS events are made based on the results of cosine similarity. Remarkably, this approach does not rely on any training data from the development dataset, external data, or pretrained models. The proposed system achieved an F1-score of 44.187% on the 2023 validation set.

Index Terms— Feature engineering, supervised learning, few-shot sound event detection.

1. INTRODUCTION

Few-shot sound event detection poses a significant challenge, representing a supervised learning problem aimed at minimizing the need for manual annotation. Essentially, it can be conceptualized as a classification problem with a temporal dimension. The overall goal is to effectively distinguish two classes, namely the POS and the NEG, and the evaluation metric is typically the F1 score.

To improve the F1 score, we utilize feature engineering, which involves automatically exploring frequency ranges of POS events within the mel-spectrogram. This algorithm examines the frequency range of the POS signal by analyzing the five preceding POS shots and up to five NEG shots.

Additionally, we experimented with the approach used in the DCASE 2022 Task 5 challenge No.1, which involved introducing a new POS event and then conducting training. This approach yielded improved performance in our experiments as well.

The remainder of this report is organized as follows: Methodology is described in Section 2, F1 scores on validation data are presented in Section 3, and discussions and conclusions are provided in Sections 4 and 5, respectively.

2. METHODOLOGY

2.1. Background knowledge and terminologies

In this section, some terminologies of this report are briefly explained first.

Mel-spectrogram

The mel-spectrogram depicts the intensity of a sound clip acrossfrequencies and time. An example is shown in Figure 1, where frequency is the vertical axis, and time is the horizontal axis. The brightness of each frequency bin corresponds to its energy or magnitude at a given time, with brighter colors indicating stronger energy levels.

Chromagram

The chromagram illustrates the distribution of energy across pitch classes (in 12 semitones), and an example is shown in Figure 2. Each bin in the chromagram corresponds to a specific pitch class and indicates the presence or strength of that pitch class at a particular time frame.

Dimension reduction

Mel-spectrogram and chromagram are two-dimensional features. In this work, we perform summation along the time and frequency axes to reduce the data dimension. Compression along the vertical axis is referred to as the time marginal distribution (TMD), while compression along the horizontal axis is termed the frequency marginal distribution (FMD). This conceptualization is inspired by Yuan et al. [1].

Cosine similarity

The formula for computing the cosine similarity between two vectors P and Q is as follows,

$$\text{Cosine similarity} = \frac{P \cdot Q}{\|P\| \, \|Q\|} \quad (1)$$

where: $P \cdot Q$ is the dot product of vectors P and Q. ||P|| and ||Q|| are the magnitudes of vectors P and Q, respectively.

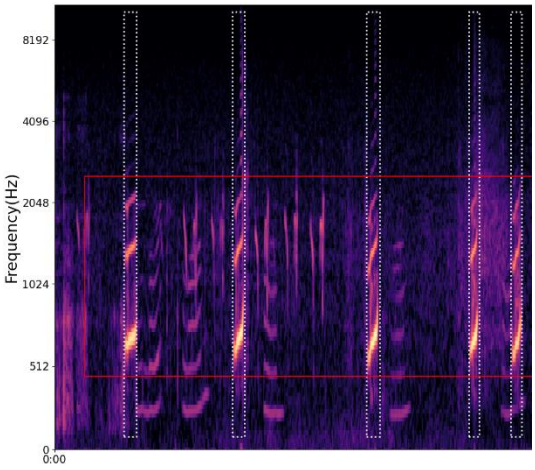

Figure 1: Mel-spectrogram of ME2.wav. The red lines mark the boundary of desired frequency range, as explained in Sec. 2.2, and the white boxes are POS shots.

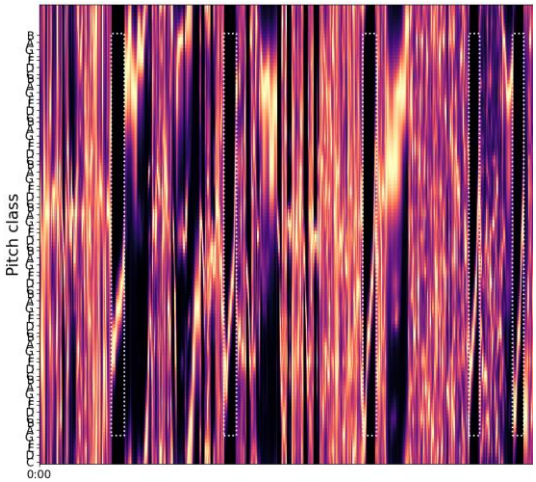

Figure 2: Chromagram of ME2.wav

2.2. Automatic Frequency Range Identification

The design logic of the proposed algorithm can be summarized as follows: each POS will search within NEG to find the most similar one in terms of cosine similarity which jointly considers TMD and FMD. Then, the FMD of the most similar NEG is subtracted from that of the POS, with negative values set to 0. After obtaining five subtraction results, the average is taken, and an example is shown in Figure 3. Then, the maximum peak is selected, and a height threshold is set for searching the left and right boundaries of the desired frequency range. When encountering values lower than the threshold, the search stops. Finally, the frequency range, i.e., the minimum frequency and maximum frequency, is determined as illustrated by the red box in Figure 1. The code for Automatic Frequency Range Identification will be uploaded to the first author's GitHub [2] after the 2024 challenge concludes.

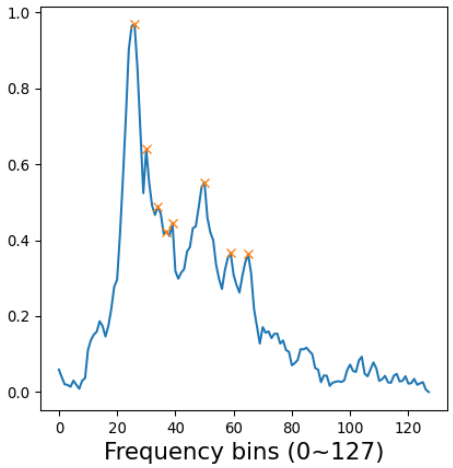

Figure 3: The average of five subtraction results for ME2.wav

2.3. Cosine similarity between POS and NEG

In the mel-spectrogram, cosine similarity is exclusively computed within the designated frequency range. As for the chromagram, bandpass filtering is applied in the time domain before generating the chromagram to ensure that we focus on the specified frequency range.

To calculate the cosine similarity between POS and NEG for the entire audio file, a window is initially defined with a length specified in Table 1, where X denotes the maximum of 5 POS durations, and Y denotes the minimum of 5 POS durations. If the window length encounters a choice between two options, the smaller window length is selected. After determining the window length, NEG shots smaller than the window length and duration of NEG shots with 0 samples are ignored and not used.

This window traverses the audio file from start to finish, advancing one frame at a time, with each frame approximately equaling 11.65 milliseconds. At each step, cosine similarity is computed in two pairs: one pair involves the window and the five POS shots, while the other pair involves the window and the maximum of five NEG shots. The calculation of cosine similarity involves TMD and FMD. Finally, the results of TMD and FMD calculations are averaged to obtain the positive cosine similarity (PCS) and negative cosine similarity (NCS), as shown in Figures 4 and 5, respectively. The green "o" represents the POS cosine similarity, while the red "x" represents the NEG cosine similarity. In Figures 4 and 5, the short horizontal lines mark the cosine similarity based on TMD, while short vertical lines mark that of the FMD.

The time complexity of this algorithm is O(n^2). To accelerate computation, we employ three steps to handle NEG events: Step 1: Truncation of excessively long NEG events -- if the duration of a NEG event falls between 90 and 400 seconds, only half of the event's duration is utilized. For NEG events with durations exceeding 400 seconds, only one-third of the duration is used. Step 2: Omission of redundant NCS values -- the search window is confined within NEG events, and if the cosine similarity between two frames is greater than or equal to 0.98, only one frame is selected for use. Step 3: Handling of UNK events within NEG events -- if more than one UNK event is present within a NEG event, the entire NEG event is excluded from consideration.

However, if only one UNK event is present within a NEG event, the duration of that UNK event is excluded while utilizing the remaining durations of the NEG event.

Table 1: window length setting

Samples	Window length
$X \leq 250$	Y
$250 < X \leq 450$	X/3 or Y
$450 < X \leq 900$	X/5.5 or Y
$900 < X$	Y

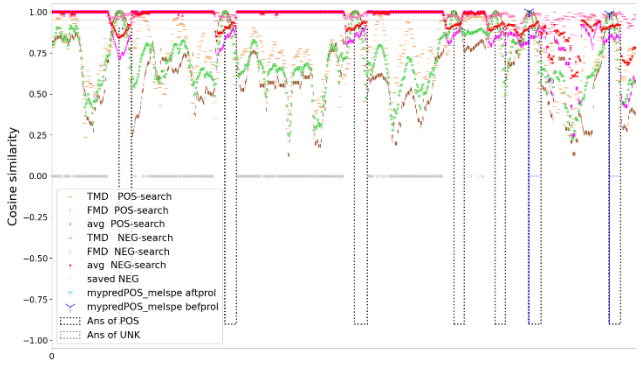

Figure 4: Cosine similarity in mel-spectrogram for ME2.wav

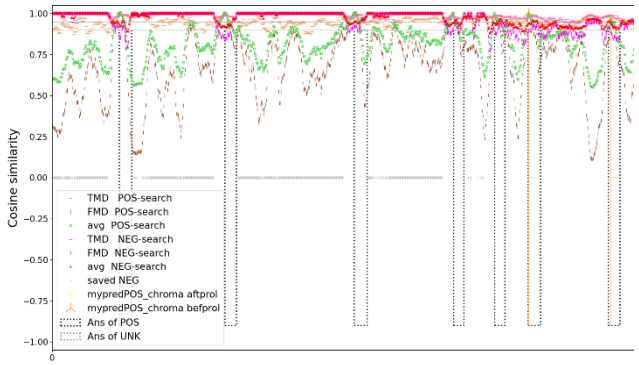

Figure 5: Cosine similarity in chromagram for ME2.wav

2.4. Predictions of POS

Based on the PCS and NCS calculation described in Section 2.3, predictions of POS events are made. Through observations, it can be noted that POS events frequently occur when the PCS is higher and the gap between PCS and NCS is larger. Following this principle, we designed our prediction system.

To introduce a new POS event and then conduct training, we designed a program that can specify the number of new POS events based on a high threshold, thereby increasing the accuracy of the new POS event. This idea can be easily implemented using the Python package scipy.signal.find_peaks, and the process is hereafter called *Findpeak*. There are two thresholds: one requires the PCS to be large and larger than NCS, and the other requires a large gap between PCS and NCS. For example, specifying 5 new POS events entails finding the top 5 largest values that match the threshold.

We developed a program called GuessnumofPOS to estimate the number of POS events in the entire audio file. This program utilizes the time durations of the first 5 shots and applies four thresholds to determine the POS count using the *Findpeak* function. The threshold conditions are as follows: Threshold 1 requires PCS > 0.95, PCS > NCS, and NCS < 0.95; Threshold 2 requires PCS > 0.9, PCS > NCS, and NCS < 0.95; Threshold 3 requires PCS > 0.9 and PCS > NCS; Threshold 4 requires PCS > 0.9. The design principle is to ensure that the estimated count is less than or equal to the actual count.

Additionally, we designed a fail-safe system to detect large patches of noise based on the slope of NCS in window lengths less than or equal to 6. When the audio duration is less than or equal to 21 minutes, we used a Savitzky-Golay filter to smooth the NCS. If the slope meets certain threshold criteria and the duration exceeds eight times the maximum length of the first five shots, the time range is considered as NEG.

The final prediction is based on the integration of two sets of parameters applied to the Findpeak function. The first set includes the minimum PCS and minimum PCS-NCS values obtained after specifying 5 or fewer POS events. The second set involves specifying the estimated POS count using GuessnumofPOS and obtaining the minimum PCS and minimum PCS-NCS values accordingly.

After identifying the location of POS events, if the ratio of maxdurationPOS to mindurationPOS for a given audio file is greater than or equal to 2, the POS event should be prolonged. This prolongation is guided by specified thresholds based on the values of PCS and NCS, extending the POS location both to the left and to the right until the thresholds are no longer met.

Merging is intended to transform non-continuous (e.g., [590, 592, 593]) prediction samples into a continuous sequence (e.g., [590, 591, 592, 593]). The merging process will only occur if three conditions are met:

1.The ratio of maxdurationPOS to mindurationPOS for a given audio file is greater than or equal to 2.

2.There is at least one negative duration sample (>= 1 sample) among the first five shots.

3.The window length is greater than or equal to 10, and the gap between two non-continuous prediction samples (e.g., 2 = 592 - 590) is less than the window length.

3. RESULT

We report the performance of our four systems on the 2023 validation set, as shown in Table 1. Our prediction system achieved an F1-score of 44.187% on the 2023 validation set. The three subsets in the validation set are categorized as long shot (HB, approximately 20 seconds), median shot (ME, approximately 1 second), and short shot (PB, approximately 0.3 seconds).

For the HB dataset, the primary issue encountered was prolonging, where multiple POS events needed to be connected into a single continuous segment, which sometimes failed. For the ME dataset, the presence of NEG events that were similar to POS events negatively impacted the F1 score. For the PB dataset, the suboptimal performance was due to the POS events in the data shifting up and down in the mel-spectrogram, leading to inaccurate predictions.

The results of automatic frequency range identification are listed in Table 2 and Table 3 in the Appendix for 2024 evaluation set and 2023 validation set, respectively. Note that the bandwidth as well as the location of the min and max bin vary greatly among the files.

Table 1: Results on the 2023 validation set

System	Precision	Recall	F1-score (%)	F1 − HB	F1 − ME	F1 − PB
Mel-spectrogram	0.41079	0.47803	**44.187**	0.61038	0.74545	0.2625
Chromagram	0.27652	0.38663	32.244	0.47324	0.35151	0.23008
Mel-spectrogram ∪ Chromagram	0.26580	0.56947	36.244	0.55005	0.39436	0.25487

4. DISCUSSION

Here, we outline five factors that influence the F1 score, listed in order:

- **Dimension Reduction Method**: The dimension reduction method used in this study involves compressing the mel-spectrogram and chromagram along the vertical and horizontal axes. This approach is less effective in distinguishing between the downward traces and upward traces in images in MGE data, which are mirror-like features. Possible solutions include changing the dimension reduction method or calculating the similarity directly using two-dimensional images without reduction. However, achieving a solution that works universally well for all data is challenging.
- **Cosine Similarity Calculation**: The method proposed in this study utilizes cosine similarity to assess how similar two verctors are However, when facing data in the Validation Set PB folder, its effectiveness is compromised because the POS events in the data tend to shift up and down in the mel-spectrogram. This leads the prediction system to misinterpret them as NEG events. Although using data augmentation to shift the first 5 POS shots up and down may mitigate this issue, this approach lacks generality and may result in inconsistent scores across different datasets. To address this problem, a program should be designed to determine for each data file whether data augmentation, specifically shifting the first 5 POS shots up and down, should be applied or not based on specific criteria.
- **Weak POS Signals**: The first 5 POS shots are occasionally weak POS signals (e.g., CHE_15.wav, DC06.wav), whereas the POS shots to be predicted are strong signals. This discrepancy can cause the prediction system to fail to identify POS shots. A potential solution is to use image enhancement or image restoration techniques to transform weak POS signals into strong POS signals. The current feasible method is to use "add pos to train" which can handle CHE_15.wav.
- **POS within NEG**: Sometimes POS events appear within NEG segments (e.g., QU06.wav), causing the PCS and NCS values to be similar. We assume that the NEG segment is the duration remaining after subtracting the duration of the first 5 POS shots from the time before the fifth POS shot. A possible solution is to compare the similarity between POS and NEG in the first 5 shots and exclude NEG durations that are too similar to the POS.
- **POS Event Duration Prediction**: The issue of predicting the duration of a POS event arises from the variability in the duration of each POS event, with some being longer or shorter than others. While it is feasible to visually determine the appropriate duration extension for most data, some datasets, like those in the CHE23 folder, contain multiple POS events clustered together. Without access to the official labeled answers, determining whether to use method A (one POS event for the entire segment) or method B (multiple POS events) poses a challenge. Each method yields different F1 scores, as the F1 score calculates the number of true positives (TP) and false positives (FP). In the furure, it may be worth introducing a new evaluation metric based on the predicted duration of POS events. This metric would assess how well the predicted durations overlap with the ground truth durations. For example, it would calculate the percentage of predicted POS durations that match the ground truth durations and the percentage that do not. A high percentage of matching durations should be given a higher score.

5. CONCLUSIONS

In this study, we developed a cosine similarity-based system for few-shot bioacoustic event detection, emphasizing automatic frequency range identification in mel-spectrograms. Our approach does not rely on any training data from the development dataset, external data, or pretrained models, demonstrating its robustness and generalizability.

Our method achieved an F1-score of 44.187% on the 2023 validation set, which includes subsets categorized as long shot (HB), median shot (ME), and short shot (PB). Each subset presented unique challenges, such as the need to prolong multiple POS events in the HB dataset, the similarity between NEG and POS events in the ME dataset, and the vertical shifting of POS events in the PB dataset.

Through comprehensive analysis, we identified several factors affecting the F1 score, as discussed in Section 4:

- **Dimension Reduction Method**: The current dimension reduction method is less effective in distinguishing mirror-like features.
- **Cosine Similarity Calculation**: Vertical shifting of POS events in mel-spectrograms led to misclassification.
- **Weak POS Signals**: Weak POS signals in the first 5 shots confounded the subsequent detection of strong POS signals.
- **POS within NEG**: POS events within NEG segments caused close similarity values.
- **POS Event Duration Prediction**: Variability in POS event durations posed particular challenges.

Despite these challenges, our system demonstrates the potential of feature engineering in bioacoustic event detection. Future work will focus on addressing identified issues and refining the system to improve its accuracy and robustness. Our code for Automatic Frequency Range Identification will be shared on GitHub after the 2024 challenge, contributing to the broader research community. In shorts, this study highlights the importance of precise feature engineering in few-shot learning scenarios, paving the way for more accurate and efficient bioacoustic event detection systems.

6. ACKNOWLEDGMENT

In tackling classification problems, employing feature engineering to enhance accuracy is crucial. Sheng-Lun Kao expresses gratitude to Professor Tai-Shih Chi of National Chiao Tung University for imparting this concept during lectures.

Sheng-Lun Kao conceived, designed, and implemented the study, as well as authored the manuscript. Yi-Wen Liu provided invaluable feedback and guidance throughout the project and contributed to the revision of the manuscript.

7. REFERENCES

[1] Yuan, S., Wang, Z., Isik, U., Giri, R., Valin, J. M., Goodwin, M. M., & Krishnaswamy, A. (2022, May). Improved singing voice separation with chromagram-based pitch-aware remixing. In Proc. *2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 111-115).

[2] https://github.com/kaoshenglun

8. APPENDIX

Here are more examples of using the 2024 evaluation set and the 2023 validation set for automatic frequency range identification. Table 2 and Table 3 respectively list the identified minimum frequency and maximum frequency for all the files in the 2024 evaluation set and the 2023 validation set's, respectively. The total number of mel bins is 128, corresponding to a highest frequency of 11025 Hz.

Table 2: Results of automatic frequency range identification on the 2024 Evaluation Set

Audio file name	**Min bin**	**Max bin**
CHE_01.wav	61	106
CHE_02.wav	17	31
CHE_03.wav	16	30
CHE_04.wav	15	31
CHE_05.wav	62	104
CHE_06.wav	58	106
CHE_07.wav	62	112
CHE_09.wav	60	121
CHE_10.wav	13	28
CHE_11.wav	13	28
CHE_12.wav	14	29
CHE_13.wav	67	103
CHE_14.wav	60	114
CHE_15.wav	11	31
CHE_16.wav	68	112
CHE_17.wav	81	105
CHE_18.wav	58	110
CHE_19.wav	68	110
CHE_F02.wav	20	76
CHE_F03.wav	15	61
CHE_F05.wav	14	77
CHE_F06.wav	30	76
CHE_F07.wav	18	77
CHE_F08.wav	44	75
CHE_F09.wav	19	84
CHE_F10.wav	18	81
CHE_F11.wav	18	76
CHE_F12.wav	18	66
CHE_F13.wav	19	76
CHE_F14.wav	36	75
CHE_F15.wav	19	67
CHE_F17.wav	15	80
CHE_F18.wav	28	68
CHE_F19.wav	15	77
ct1.wav	78	113
ct2.wav	2	23
ct3.wav	97	123
cw1300_DCASE.wav	4	35
cw1315_DCASE.wav	5	59
cw1330_DCASE.wav	2	45
cw1345_DCASE.wav	4	47
DC01.wav	17	31
DC02.wav	19	31
DC04.wav	17	34
DC05.wav	11	29
DC06.wav	15	41
DC07.wav	68	106
DC08.wav	81	125
DC10.wav	83	119
DC11.wav	89	122
DC12.wav	78	110
85MGE.wav	67	86
89MGE.wav	77	113
91MGE.wav	68	90
E1_208_20190712_0150.wav	54	108
E2_208_20190712_0150.wav	54	86
E3_49_20190715_0150.wav	48	95
E4_49_20190804_0150.wav	56	104
QU01.wav	31	46
QU02.wav	6	36
QU03.wav	37	63
QU04.wav	16	29
QU05.wav	9	46
QU06.wav	7	26
QU07.wav	12	27
QU08.wav	100	114

Table 3: Same as Table 2, tested on the 2023 Validation Set

Audio file name	**Min bin**	**Max bin**
file_423_487.wav	31	56
file_97_113.wav	3	23
R4_cleaned recording_13-10-17.wav	3	23
R4_cleaned recording_16-10-17.wav	3	23
R4_cleaned recording_17-10-17.wav	6	42
R4_cleaned recording_TEL_19-10-17.wav	6	75
R4_cleaned recording_TEL_20-10-17.wav	12	42
R4_cleaned recording_TEL_23-10-17.wav	13	43
R4_cleaned recording_TEL_24-10-17.wav	13	38
R4_cleaned recording_TEL_25-10-17.wav	2	23
ME1.wav	17	33
ME2.wav	17	73
BUK1_20181011_001004.wav	100	115
BUK1_20181013_023504.wav	107	123
BUK4_20161011_000804.wav	107	122
BUK4_20171022_004304a.wav	100	119
BUK5_20161101_002104a.wav	88	125
BUK5_20180921_015906a.wav	100	121