A novel approach using deep belief network patterns and attention binary decomposition for automated community emotion detection

Biomedical Signal Processing and Control 116 (2026) 109534

Contents lists available at ScienceDirect

Biomedical Signal Processing and Control

journal homepage: www.elsevier.com/locate/bspc

A novel approach using deep belief network patterns and attention binary decomposition for automated community emotion detection

Arif Metehan Yildiz [a], Prabal Datta Barua [b], Mehmet Baygin [c], Sengul Dogan [a], Turker Tuncer [a], Massimo Salvi [d,*], Ru-San Tan [e,f], U.R. Acharya [g]

[a] Department of Digital Forensics Engineering, College of Technology, Firat University, 23119 Elazig, Turkey
[b] School of Business (Information System), University of Southern Queensland, Australia
[c] Department of Computer Engineering, College of Engineering, Erzurum Technical University, Erzurum, Turkey
[d] Department of Electronics and Telecommunications, Politecnico di Torino, Torino, Italy
[e] National Heart Centre Singapore, Singapore
[f] Duke-NUS Medical School, Singapore
[g] School of Mathematics, Physics and Computing and Centre for Health Research, University of Southern Queensland, Springfield, Australia

ARTICLE INFO

Keywords:
DBNPat
Attention binary pattern decomposition
SCED
Sound forensics
Environmental sound classification

ABSTRACT

Context: Sound-based community emotion detection (SCED) estimates community emotion from environmental sounds. It has value for public safety and human–computer interaction. Current SCED models have limited adaptivity on complex audio and often need manual tuning.
Objective: We aim to design an accurate and efficient automated SCED model for large-scale data.
Methods: We propose a feature extraction framework that combines DBNPat feature generation with ATT-BP attention-driven binary compression. The framework adapts to signal characteristics with low computational cost. We also introduce a new dataset of 10,017 environmental sound clips (three seconds) with negative (n = 1,729), neutral (n = 6,154), and positive (n = 2,134) classes.
Results: The proposed SCED model achieves 87.28% accuracy on three-class SCED. It yields 81.30% UAR, 84.71% precision, 82.97% F1, and 80.59% geometric mean on the imbalanced dataset. *Conclusion:* The model links classical feature design and deep pattern generation in one adaptive pipeline. It offers a practical solution for digital sound forensics and other ambient-audio systems that need fine emotion cues.

1. Introduction

Sound-based community emotion detection (SCED) links sound forensics and digital forensics and forms a growing research area [1,2]. The SCED models predict emotional states from environmental audio and supports applications in public safety, social analysis, and human–computer interaction [3]. SCED treats ambient sound as a direct source of emotion-related information in shared environments [4]. It supplies affective cues that text-based and image-based methods cannot capture [5,6].

Emotion is a central element of human communication and decision processes [7,8]. Emotion arises from biological and psychological processes and holds a central role in human experience [9]. The ability to detect and interpret emotions remains critical for communication and decision tasks [10]. Emotion recognition/classification techniques/models contribute to human–computer interaction (HCI) and establish new opportunities for interpreting affective states [11–13]. These techniques (emotion recognition models) support many fields, and these fields are social media analysis, market research, clinical evaluation of emotional disorders, and community emotion studies, digital forensics especially sound forensics and information security and these models have used visual or auditory datasets [14]. Thus, these models also support analysis of emotional states in sound-based conditions [15–17].

The emotion recognition problem is a nondeterministic polynomial problem and machine learning is a best way to solve these types problems since by utilizing machine learning, emotions patterns have been detected using data [18]. This connection provides systematic interpretation of human emotions based on sound characteristics [19].

* Corresponding author.
E-mail addresses: 211144202@firat.edu.tr (A.M. Yildiz), Prabal.Barua@usq.edu.au (P.D. Barua), mehmet.baygin@erzurum.edu.tr (M. Baygin), sdogan@firat.edu.tr (S. Dogan), turkertuncer@firat.edu.tr (T. Tuncer), massimo.salvi@polito.it (M. Salvi), Rajendra.Acharya@usq.edu.au (U.R. Acharya).

https://doi.org/10.1016/j.bspc.2026.109534
Received 22 April 2025; Received in revised form 26 November 2025; Accepted 8 January 2026

Machine learning models extract features, select the most informative features and classify emotional states in environmental audio and supply insights into community behavior [20].

1.1. Related works

The most known research about SCED is the model by Yildiz et al. [18], who introduced a sound-based framework for community emotion analysis. Their study aims to detect emotional states in audio and uses a pattern-based model. Their approach processes binary temporal patterns and assigns emotional categories across time, and the authors report strong performance for community emotion detection. Given that SCED is a relatively new research area with limited dedicated models, we also reviewed related work in individual emotional recognition and environmental sound classification.

Existing sound classification datasets illustrate the current limitations in the field. The ESC-50 dataset contains only 2,000 clips and focuses on sound events like “dog barking” and “drilling” rather than emotional content [21]. Similarly, UrbanSound8K includes 8,732 clips but emphasizes sound event classification [22]. Speech emotion datasets such as RAVDESS [23] and EMO-DB [24] contain fewer than 3,000 recordings and are limited to individual speakers. These constraints highlight the need for a large, diverse dataset specifically designed for SCED research.

In this study, we define “Community Emotion” as the emotional atmosphere shared by groups occupying a physical space. This definition encompasses emotional conditions observed in environmental sound scenes, including stadium crowds, protest environments, and quiet public spaces. It differs from online text-based sentiment analysis and single-speaker speech emotion recognition by focusing on group-level emotional conditions.

1.2. Limitations

SCED is important for public security, information security and digital forensics since it is a real-life problem, yet most studies focus on Speech Emotion Recognition (SER) or Environmental Sound Classification (ESC). Our survey shows several limits in current SCED work. Self-organized approaches are rare, even though they can adapt to complex audio without heavy parameter control. Feature extraction methods that follow structural ideas from deep learning networks are also missing. Existing SCED datasets are small and do not include a wide range of emotional cases. These limits reduce the generalization of SCED systems and restrict their use in real settings.

1.3. Motivation and main contributions

We aim to develop an efficient and accurate automated SCED model on a large sound dataset. The model is designed to remain lightweight while offering feature extraction performance close to deep learning. The design takes inspiration from Transformers and Deep Belief Networks (DBNs). A feature engineering framework is proposed. This structure removes the need for heavy models. Signal decomposition and graph-based textural feature extraction are used to address limits in existing methods. The approach is evaluated on a dataset with more than 10,000 labeled sound segments that represent negative, neutral, and positive emotional states.

1.4. Contributions

- We curated a new dataset with 10,017 audio signals, equal to 524 min of sound, and classify these signals into three emotion classes. This dataset reduces the lack of large SCED resources.
- We develop an attention-inspired binary pattern method named ATT-BP. The method creates multiple compression bands from raw audio and avoids routing problems that appear in standard compression functions.
- A deep belief network pattern textural extractor named DBNPat has been presented in this research. This extractor follows a lattice form based on DBN layers and supports self-organized feature selection.
- A multilevel hybrid feature engineering framework has been recommended in this work. The system combines statistical and nonlinear textural features and adapts to properties of the input signals.
- The recommended feature engineering model yields 87.28 % accuracy on three emotion classes in a large and imbalanced dataset with linear time complexity.

2. Materials

The curated dataset in this study comes from environmental audio segments taken from the YouTube-8 M dataset (https://research.google.com/youtube8m). YouTube-8 M is a public dataset created by Google Research and this dataset was curated for scientific researches. It contains millions of videos and includes more than 3,800 audio–visual entity labels. These labels provide a structured and reproducible foundation for data collection.

The curated SCED dataset does not depend on random keyword searches on YouTube. It uses the predefined entity labels in YouTube-8 M. These labels place videos into clear acoustic groups such as Concert, Applause, Protest, Traffic, and Rain. This approach keeps the selection process systematic and consistent. It also secures full reproducibility for future studies.

2.1. Entity-based filtering and clip extraction

The entity taxonomy in YouTube-8 M was first reviewed, and categories with strong environmental sounds relevant to community emotion analysis were identified. These entities were then grouped into three emotion-related classes under the Valence–Arousal framework. Positive-related entities were placed under Concert, Music festival, Applause, Party, Ceremony, Dance, Cheerleading, and Comedy (laughter). Negative-related entities were assigned to Protest, Riot, War, Combat, Police siren, Explosion, Accident, Fire, Emergency vehicle, and Funeral. Neutral-related entities were listed as Traffic, Street ambience, Office, Public transport, Rain, Wind, Forest, and Construction noise.

Videos that carried at least one of these entity labels were selected for processing. Audio streams were obtained in WAV, MP3, and MPEG formats and at sampling rates between 22.05 and 48 kHz. All audio files were converted to a uniform format before segmentation. Each track was divided into three-second clips, following common practice in environmental sound research. A total of 10,017 labeled clips were produced, and these clips formed the dataset, and we applied our presented feature engineering model to get classification results.

2.2. Annotation procedure

YouTube-8 M includes semantic entity labels, but emotional labels are not provided since emotional classes were assigned with a structured multi-annotator protocol. Each three-second audio segment was examined by five independent annotators. The related video frames from the YouTube-8 M metadata were also reviewed.

The annotation procedure was carried out in two stages. In the first stage, an emotion label (negative, neutral, or positive) was assigned independently by all annotators. A majority voting rule with a minimum agreement of three out of five was applied, and an initial label was obtained for each clip. In the second stage, clips with differing decisions were re-evaluated in a consensus session. Final labels were assigned through a strict consensus rule that required full agreement among all five annotators.

The utilized two-stage procedure reduced individual subjectivity and

supported consistent labeling. It also formed a clear link between entity meaning, acoustic patterns, and emotional valence. The final class distribution is shown in Table 1.

Neutral ambient sounds occur more often than emotionally strong sounds. The negative class contains 1,729 clips with 93.80 min of audio and a mean length of 3.26 s. The neutral class contains 6,154 clips with 318.82 min of audio and a mean length of 3.11 s. The positive class contains 2,134 clips with 111.48 min of audio and a mean length of 3.13 s.

A total of 10,017 selected clips are included, with a total duration of 524.11 min. Only the audio tracks were used. All these steps are also summarized in Fig. 1.

2.3. Diversity assessment

The selected segments were obtained from more than 2,000 YouTube channels. A wide range of acoustic settings was included, such as human gatherings, natural ambience, transportation sounds, celebrations, and emergency scenes. The distribution of the acoustic categories is listed below:

- Environmental and nature ambience: 31.4 %
- Human crowd and mixed ambience: 26.2 %
- Events and celebrations: 22.5 %
- Incidents, emergencies, and conflicts: 10.8 %
- Transportation and machinery: 9.1 %

Spoken words and contextual clues in the videos allowed approximate language identification. The dataset includes 58 % English, 26 % Turkish, and 16 % content in other languages such as Spanish, French, and German. These values show that the dataset covers many sound sources and sociocultural settings. The diversity increases its suitability for SCED research.

2.4. Ethical and reproducibility statement

While direct dataset sharing is restricted by YouTube's licensing terms, reproducibility is ensured through comprehensive documentation of our methodology. The manuscript provides detailed entity selection criteria, preprocessing parameters, and annotation protocols. Any researcher can recreate a functionally equivalent dataset by following these procedures.

The dataset contains only environmental sounds and non-identifiable crowd noise, explicitly excluding private speech or personally identifiable content. All analyses were conducted locally for academic research purposes, maintaining full compliance with copyright and privacy regulations.

3. Methods

3.1. Proposed model architecture

Our presented feature engineering model's main objective is to obtain high emotion classification performance with a linear time complexity. This model is organized into four sequential phases, as shown in Fig. 2. In the first phase, multilevel hybrid feature extraction is applied. The ATT-BP signal decomposition method is used, and statistical features and DBNPat features are also obtained. In the second phase, feature selection is performed. Neighborhood component analysis (NCA) [25], Chi-square (Chi2) [26], and minimum redundancy maximum relevance (mRMR) [27] are used to identify the most informative features.

The third phase (classification) applies classification. k-nearest neighbors (kNN) [28] and support vector machine (SVM) [29] classifiers are used. In the final phase, information fusion is carried out. Iterative majority voting (IMV) [30] and a greedy algorithm [31] are applied to produce voted outputs and to determine the final classification result. The graphical block diagram of the DBNPat-based model is showcased in Fig. 2.

Our proposed SCED classification pipeline begins with the raw sound signal. The signal is decomposed by the ATT-BP compression function into 11 separate compression bands. These bands and the raw signal are then passed in parallel through the DBNPat-based textural extractor and the statistical extractors. The recommended DBNPat is a feature extraction version of a deep learning models. By utilizing these feature extraction functions (DBNPat and statistics), 768 textural and 40 statistics of these textural features are obtained, and 40 direct statistical features are computed. In this aspect, 848 features are generated for each input and we have used 12 inputs for feature generation. Therefore, the length of final feature vector is 10,176 (=848 × 12).

Table 1
Attributes of the collected SCED dataset.

Class	Number of sounds	Percentage (%)	Time (total/average)
Negative	1729	17.26	93.80 min/ 3.26 s
Neutral	6154	61.44	318.82 min/ 3.11 s
Positive	2134	21.30	111.48 min / 3.13 s
Total	10,017	100	minutes / 3.14 s

3.2. Feature extraction

3.2.1. Signal decomposition using ATT-BP

The presented first method for this paper is ATT-BP. Herein, we have inspired from transformers to recommended this decomposition method. In this method, binary pattern (BP) transformation [32], maximum absolute pooling (it is our designed pooling function), and average pooling [33] were applied to compute multiple compression bands. To better explain this model, the graphical depiction of the introduced ATT-BP is showcased in Fig. 3.

The ATT-BP algorithm begins with the division of the raw audio signal into overlapping blocks of length 9. A stride value of 2 is used for this step. A binary pattern transformation is then applied to each block. Every sample in the block is compared with the central sample at position 5. This comparison produces an 8-bit pattern, as defined in Equation (1). In this equation, bit(g) denotes a binary feature, and σ^1 is the signum function that compares each sample bl(a) with the central sample bl(5).

$$bit(g) = \sigma^1(bl(a), bl(5)), a \in \{1,2,\cdots,9\}, a \neq 5, g \in \{1,2,\cdots,8\} \quad (1)$$

These binary features are then converted into a decimal map value B:

$$B = \sum_{g=1}^{8} bit(g) \times 2^{8-g} \quad (2)$$

The maximum absolute value (M) in the block is computed deploying maximum absolute pooling, as shown in Equation (3). The average value (A) of the block is also computed, as given in Equation (4).

$$\begin{aligned} id &= \max(|bl|) \\ M &= bl(id) \end{aligned} \quad (3)$$

$$A = mean(bl) \quad (4)$$

The final compressed value (P) is computed deploying Equation (5).

$$P = A + B \times M \quad (5)$$

This process is applied to all signal blocks, and a compressed form of the original signal is produced. The ATT-BP function is then applied again in a recursive manner. Eleven compression bands are generated in total. Each band is derived from the band produced in the previous step.

$$band^1 = att\_bp(sound) \quad (6)$$

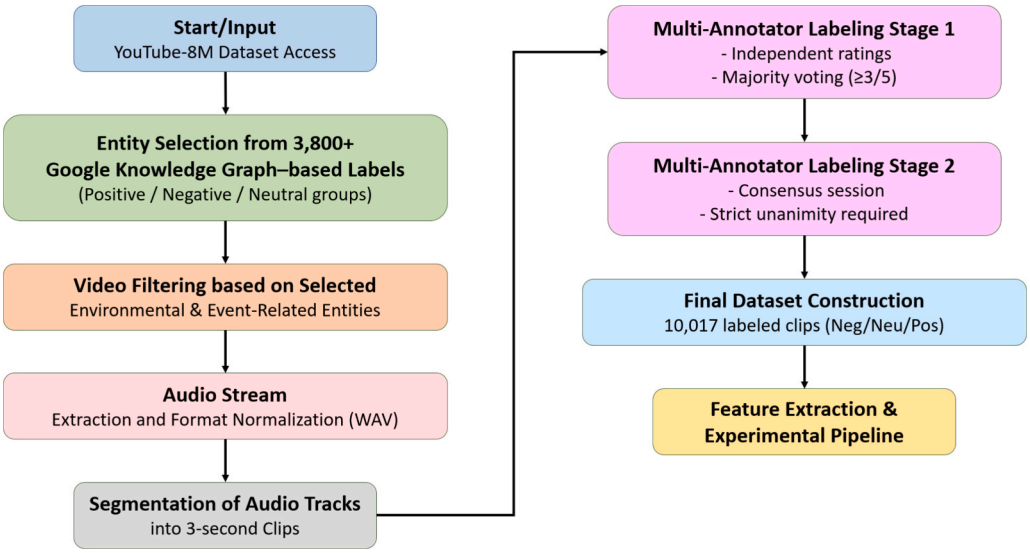

Fig. 1. Data acquisition and processing pipeline.

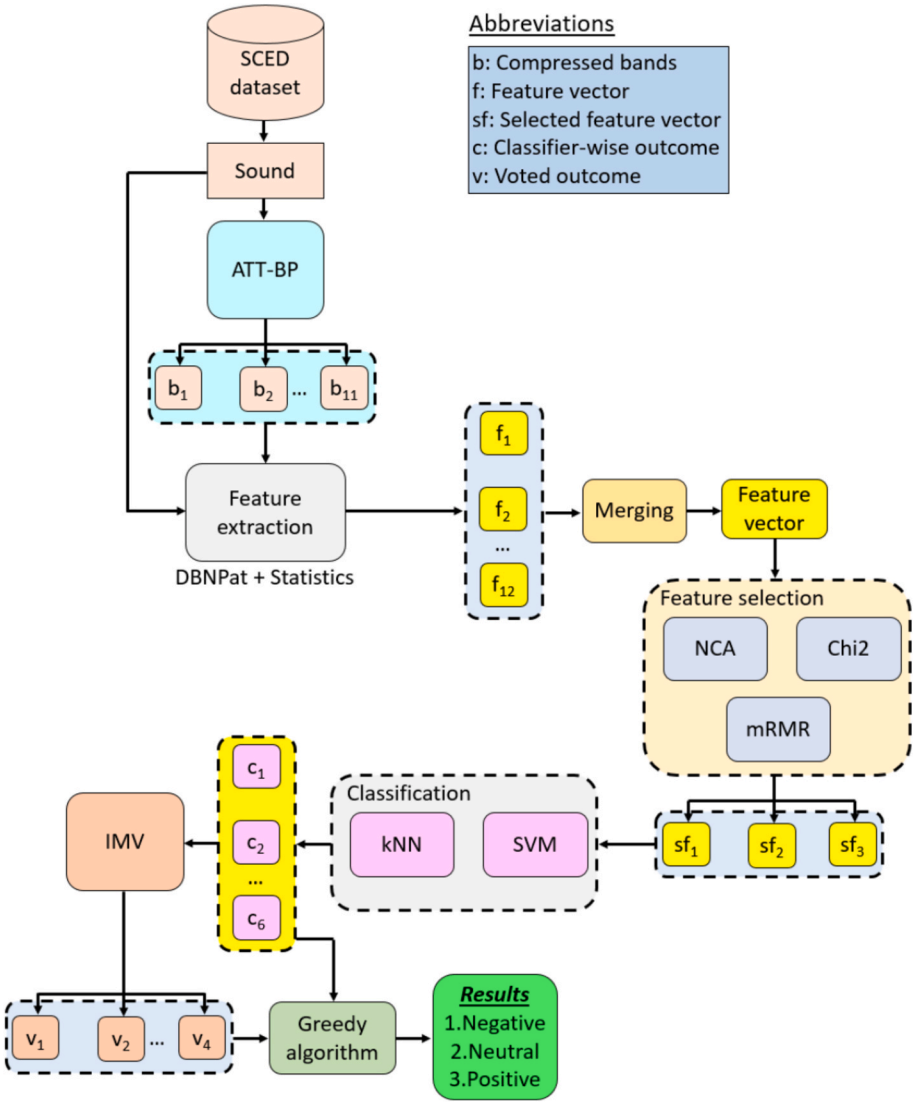

Fig. 2. Our proposed SCED model.

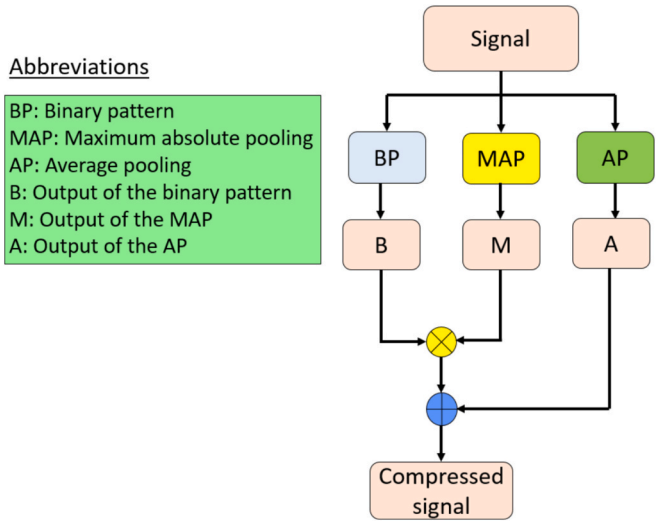

Fig. 3. Illustration of ATT-BP compression function.

it is categorized as a lattice-based feature extractor. This method utilizes the shape of a deep belief network [35] to create patterns for the data block in use, thus classifying it as a self-organized approach. In this way, it allows for the selection of the most suitable feature pattern according to the specific data block. The DBNPat algorithm first calculates a threshold value (tr) for binary feature generation using the standard deviation of the signal divided by 2, which helps in normalizing the sensitivity of the feature extraction process. The signal is then divided into overlapping blocks of length 28, which are mapped onto the DBNPat lattice structure shown in Fig. 4:

$$bl^i = signal(i+j-1), i \in \{1,2,\cdots,n-27\}, j \in \{1,2,\cdots,28\} \tag{8}$$

For each block, we compute the distances between the average value of the signal and each sample in the block:

$$db = \left| mean(signal) - bl^i \right| \tag{9}$$

Using these distances, we create a pathway through the lattice by selecting the minimum distance values within each layer of the network. This mimics the feed-forward pathway creation in a deep belief network:

$$\begin{gathered} w(1) = \min(db\{1,2,3,4\}),\ w(2) = \min(db\{5,6\}),\ w(3) = \min(db\{7,8,9,10\}), \\ w(4) = \min(db\{11,12\}),\ w(5) = \min(db\{13,14,15,16\}),\ w(6) = \min(db\{17,18\}), \\ w(7) = \min(db\{19,20,21,22\}),\ w(8) = \min(db\{23,24\}), \\ w(9) = \min(db\{25,26,27,28\}) \end{gathered} \tag{10}$$

$$band^{q+1} = att\_bp(band^q), q \in \{1,2,\cdots,10\} \tag{7}$$

3.2.2. Statistical feature extraction

To generate the statistical features, we used nonlinear entropy features and traditional statistical measures. The utilized entropy moments are: Tsallis entropy, Renyi entropy, Shannon entropy, Norm entropy, log entropy, sure entropy, wavelet entropy, and threshold entropy. The statistical features are maximum value, minimum value, variance, mode, average, kurtosis, sum of square error, range, energy, fractal dimension, and Higuchi fractal dimension [34].

These statistical features are extracted from both the input signals (raw and compressed) and from the textural features created by the DBNPat method described in the following section. This dual approach to statistical feature extraction ensures a comprehensive characterization of the signal's properties at multiple levels of analysis.

3.2.3. Textural feature extraction using deep belief network pattern

The introduced DBNPat is a textural feature extraction function, and

A visual example of the path creation is also demonstrated in Fig. 5 to better clarify the proposed self-organized pattern selection model.

With the pathway established, we extract binary features using three different kernel functions (signum, lower ternary, and upper ternary) by comparing consecutive nodes along the pathway:

$$bft^h(t) = \sigma^h\left(bl^i(w(t)), bl^i(w(t+1))\right), h \in \{1,2,3\}, t \in \{1,2,\cdots,8\} \tag{11}$$

where the three kernel functions are defined as follows:

$$\sigma^1\left(bl^i(w(t)), bl^i(w(t+1))\right) = \begin{cases} 0, bl^i(w(t)) - bl^i(w(t+1)) < 0 \\ 1, bl^i(w(t)) - bl^i(w(t+1)) \geq 0 \end{cases} \tag{12}$$

$$\sigma^2\left(bl^i(w(t)), bl^i(w(t+1))\right) = \begin{cases} 0, bl^i(w(t)) - bl^i(w(t+1)) \geq -tr \\ 1, bl^i(w(t)) - bl^i(w(t+1)) < -tr \end{cases} \tag{13}$$

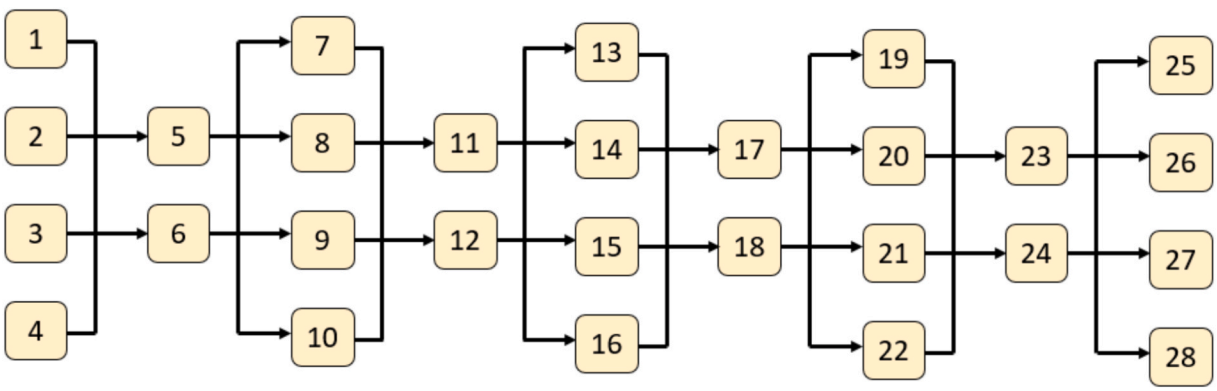

Fig. 4. Deep belief network-based lattice used in this work.

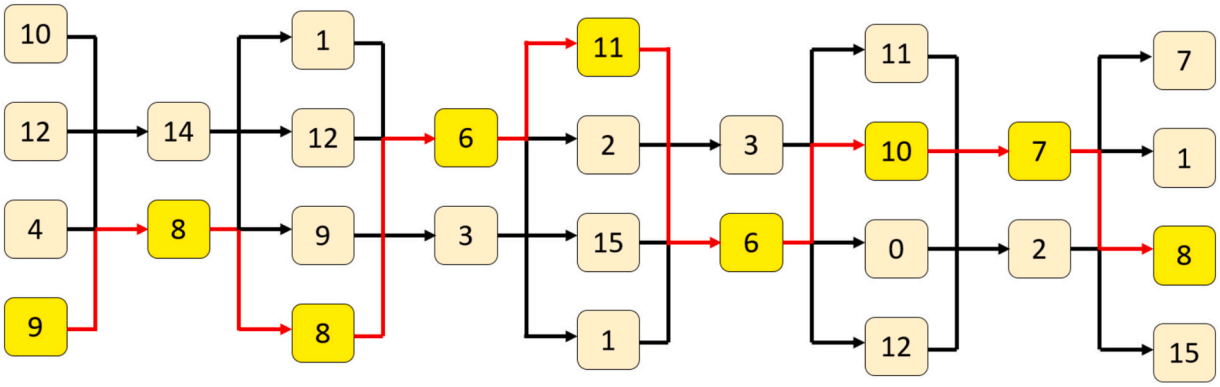

Fig. 5. Visual example of the automated pattern (path) creation model. The paths established are indicated by red arrows, while the values selected for these paths are highlighted using yellow boxes.

$$\sigma^3\left(bl^i(w(t)),bl^i(w(t+1))\right)=\begin{cases}0, bl^i(w(t))-bl^i(w(t+1))\le tr\\ 1, bl^i(w(t))-bl^i(w(t+1))>tr\end{cases} \quad (14)$$

The generated binary features are transformed into decimal values to create three map signals:

$$map^h(i)=\sum_{j=1}^{8} bf^h(j)\times 2^{8-j} \quad (15)$$

From these map signals, we extract histograms (H), each with a length of 256 (corresponding to the 2^8 possible values of an 8-bit pattern):

$$H^h=\rho\left(map^h\right) \quad (16)$$

Finally, the three histograms are concatenated to create a feature vector with length 768:

$$ftv(r+256\times(h-1))=H^h(r), r\in\{1,2,\cdots,256\} \quad (17)$$

3.2.4. Complete feature extraction process

At the end of this process, each feature vector has a length of 848, comprising 768 textural features, 40 statistical features, and 40 statistics of the textural features. These 12 feature vectors (one for the raw signal and one for each of the 11 compression bands) are concatenated to form a final feature vector (FV) of length 10,176:

$$FV=\mu\left(f^1,f^2,\cdots,f^{12}\right) \quad (18)$$

This process is repeated for all sound signals in the dataset to create the complete feature matrix (X), which serves as input to the feature selection phase.

3.3. Feature selection

The feature space produced in the previous step is high-dimensional and includes redundant or weak features (there are 10,176 features and this feature set contains redundant feature). To manage this, three established selection methods were used: NCA [25], Chi2 [26], and mRMR [27]. Each selector identified the most informative 256 features based on its own rules.

NCA assigned weights to features by maximizing expected leave-one-out accuracy with a regularization term. Chi2 measured the relationship between each feature and the class labels through the chi-square statistic. mRMR selected features that carried strong information about the classes while reducing overlap between features.

Qualified indexes of the produced features were calculated for each selector:

$$idx^h=fs^h(X,y), h\in\{1,2,3\} \quad (19)$$

where idx represents the qualified index of the features (the length is 10,176); y, actual label; fs, feature selection function, with fs^1, fs^2, and fs^3 being NCA, Chi2, and mRMR, respectively. Then, we select the most informative 256 features based on qualified indexes generated:

$$sf^h(dm,r)=X\left(dm,idx^h(r)\right), r\in\{1,2,\cdots,256\}, dm\in\{1,2,\cdots,NoS\} \quad (20)$$

where sf represents the selected feature vector, and NoS is the number of sound signals in the dataset.

3.4. Classification

Two standard shallow classifiers were used in the DBNPat-based model. kNN [28] is a distance-based method, and SVM [29] is widely used in machine learning. Both classifiers were examined in the MATLAB Classification Learner environment, and they produced the two highest baseline results. For this reason, they were selected for the study.

No hyperparameter optimization was applied. The kNN classifier used k = 1, equal weighting, and the L1-norm distance. The SVM classifier used the cubic kernel. These settings were taken directly from the MATLAB Classification Learner results.

Each classifier was applied to the three selected feature sets. This produced six classification models in total (= 3 × 2):

$$c^h=kNN\left(sf^h,y\right), \quad (21)$$

$$c^{h+3}=SVM\left(sf^h,y\right) \quad (22)$$

All models are evaluated using 10-fold cross-validation to ensure robust performance estimation.

3.5. Information fusion and selection of optimal results

The final phase of the DBNPat-based model is information fusion and this phase uses IMV [30] and greedy algorithm [31] together. IMV uses the outputs of several classifiers and creates ensemble results that may offer higher accuracy than any single classifier. The accuracy of the six classifier outputs is calculated first and sorted from highest to lowest. Voted results are then created with the mode function applied to different classifier groups. The process starts with the two most accurate classifiers and continues by adding the next classifier in the ranking. Four voted results are produced in total, based on combinations of 3, 4, 5, and all 6 classifiers. This procedure gives ten possible outcomes: six individual classifier results and four voted results. The final decision is obtained by computing the accuracy of all ten outcomes and selecting the one with the highest value.

4. Results

The introduced DBNPat-based framework was implemented in MATLAB (2023a) and executed on a standard personal computer with 32 GB RAM, a 3 GHz processor, and the Windows 11 operating system. Several evaluation metrics were used to assess performance. These metrics included classification accuracy, unweighted average recall (UAR), unweighted average precision (UAP), the overall F1-score, and the geometric mean [36]. Given our mildly imbalanced study dataset, the F1-score and geometric mean metrics provide particularly valuable assessments of performance as they account for class imbalance. The mathematical expressions for these evaluation metrics are given in Equations 23–27.

$$Acc = \frac{tp + tn}{tp + tn + fp + fn} \tag{23}$$

$$UAR = \frac{1}{C}\sum_{i=1}^{C}\frac{tp_i}{tp_i + fn_i} \tag{24}$$

$$UAP = \frac{1}{C}\sum_{i=1}^{C}\frac{tp_i}{tp_i + fp_i} \tag{25}$$

$$F1 = 2x\frac{PrecisionxRecall}{Precision + Recall} \tag{26}$$

$$GM = \sqrt{SensitivityxSpecificity} \tag{27}$$

In these equations, *tp*, *tn*, *fp*, and *fn* represent the count of true positives, true negatives, false positives, and false negatives, respectively, derived from the confusion matrix. The variable *C* denotes the total number of classes in the dataset, while *i* serves as the index for the specific class being evaluated. Additionally, in Eq. (27), *Sensitivity* corresponds to the recall rate ($tp/(tp + fn)$), and *Specificity* refers to the true negative rate ($tn/(tn + fp)$).

4.1. Classifier-specific results

The introduced DBNPat-based feature engineering framework produced high accurate results across all classifier and feature selector combinations. A maximum accuracy of 86.73 % was obtained when the NCA selector was paired with the SVM classifier (Table 2).

These results (see Table 2) showcase differences between the feature selectors and the classifiers. NCA produced higher values than Chi2 and mRMR for all metrics, and this pattern was observed with both SVM and kNN. SVM also reached higher accuracy than kNN when the same selector was used. The smallest gap between the classifiers appeared when NCA was applied (Fig. 6).

Fig. 6a demonstrates that NCA reached a mean accuracy of about 86.3 %. Chi2 and mRMR reached about 81.2 % and 81.4 %. This gap of nearly 5 % points indicates that NCA identified the most useful features for the SCED task. Fig. 6b illustrates that SVM performed better than kNN by an average margin of about 0.7 % points.

Table 2
Classifier-wise results by feature selector and classifier. The best results are highlighted in bold font.

Feature selector	Classifier	Accuracy (%)	UAR (%)	UAP (%)	F1-score (%)	Geometric mean (%)
NCA	kNN	85.95	79.31	82.71	80.97	78.59
Chi2	kNN	80.64	70.75	75.82	73.20	68.93
mRMR	kNN	81.32	71.85	75.99	73.86	70.23
NCA	SVM	**86.73**	**80.61**	**83.74**	**82.14**	**79.94**
Chi2	SVM	81.71	72.71	76.81	74.70	71.22
mRMR	SVM	81.46	72.48	75.89	74.15	71.05

4.2. Voted results

In the fusion phase, we applied IMV to the six classifier-specific outcomes to generate four distinct voted outcomes. Each voted outcome was generated using a different number of classifier-specific outcomes, starting with the three best-performing classifiers and incrementally adding the next best classifier. The results of these evaluations are presented in Table 3.

As observed in Table 3, increasing the number of classifier-wise results used (from 3 to 6) led to a consistent decrease in accuracy, UAR, and F1-score. This trend suggests that not all classifiers contribute equally to the ensemble performance, and including lower-performing classifiers can actually degrade the overall system performance. The highest performance across all metrics was achieved with the combination of the top three classifier-specific outcomes, which included NCA + SVM, NCA + kNN, and Chi2 + SVM (in descending order of individual accuracy).

4.3. Final results of the proposed model

The greedy algorithm autonomously selected the final result with the maximum accuracy from among all classifier-wise and voted results. The optimal result was the first voted result (combining the top three classifiers), which achieved 87.28 % classification accuracy, 81.30 % UAR, 84.71 % UAP, 82.97 % F1-score, and 80.59 % geometric mean. This represents an improvement of 0.55 percentage points in accuracy over the best individual classifier (NCA + SVM). The confusion matrix is shown in Fig. 7.

4.4. Feature analysis

Unlike standard deep models, our model architecture allows us to establish the relative contributions of various model elements to the final results through analysis of the provenance of selected features. For every input signal, multilevel hybrid feature extraction generates 10,176 features (12 inputs × (768 textural features + 40 statistics of textural features + 40 statistical features)), from which the three feature selectors select 768 (256 × 3) of the most informative features.

Fig. 8 reveals that the ATT-BP compression bands collectively contributed 435 selected features (56.6 %), while the raw sound signal contributed 333 selected features (43.4 %). This distribution underscores the efficacy of our ATT-BP compression approach for enhancing feature diversity and capturing deep sound characteristics. It's worth noting that the raw signal alone contributed the largest number of features from any single input source, highlighting its importance despite the value added by the compression bands.

Fig. 9 showcases the distribution of the selected features. 661 of the 768 selected features (86.1 %) were textural features produced by DBNPat. This result indicates that DBNPat extracted most of the useful information from the sound data. The remaining features were divided into two small groups: 43 direct statistical features (5.6 %) and 64 statistics of the DBNPat features (8.3 %).

The 661 textural features were also examined by kernel type. The lower ternary function produced 294 features (44.5 %). The signum function produced 261 features (39.5 %). The upper ternary function generated 106 features (16.0 %). These values show that all three kernels contributed meaningful information. The lower ternary and signum functions provided the strongest support for this SCED task.

5. Discussions

Deep learning models are widely known for strong classification performance, often higher than traditional feature engineering models [37,38]. These models extract detailed signal features automatically and remove the need for manual feature design. Therefore, deep learning models have generally exponential time burden. In contrast, classical

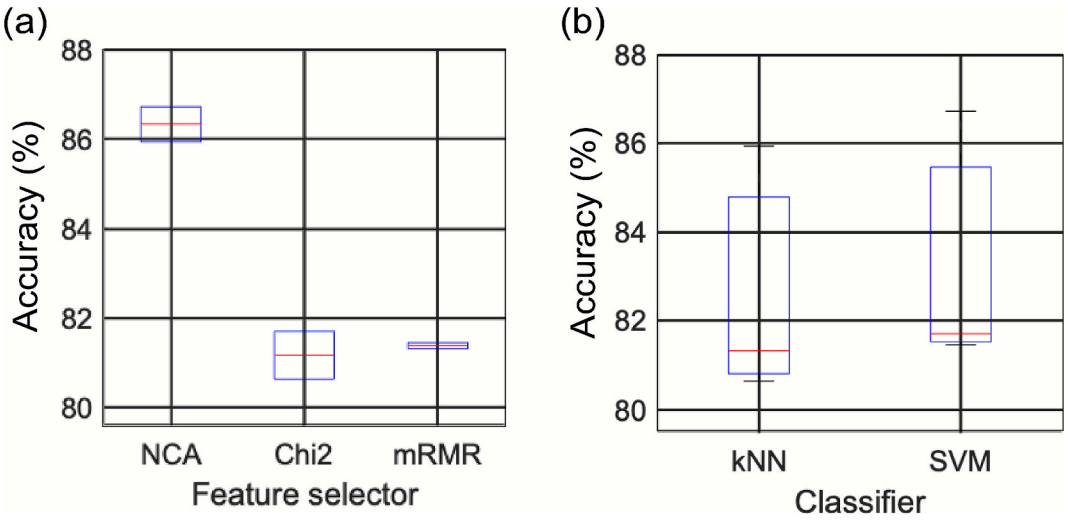

Fig. 6. (a) Mean accuracy results versus feature selection and (b) Mean accuracy versus classifiers.

Table 3
Voted results by iteration (start iteration at 3). The best results are highlighted in bold font.

Top classifier-wise results used	Accuracy (%)	UAR (%)	UAP (%)	F1-score (%)	Geometric mean (%)
3	**87.28**	**81.30**	**84.71**	**82.97**	**80.59**
4	86.92	80.76	84.94	82.80	79.79
5	86.60	79.89	84.39	82.08	79.03
6	86.41	79.84	84.29	82.01	78.90

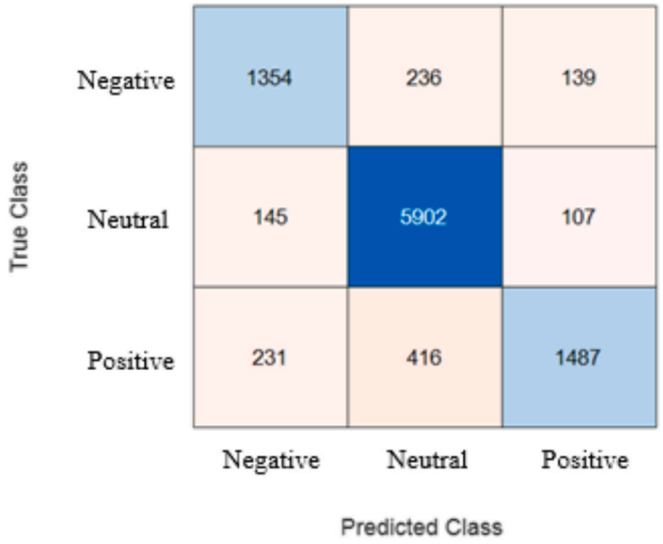

Fig. 7. Confusion matrix of final results of the proposed model.

feature engineering depends on predefined features based on fixed rules and domain knowledge. This dependence limits adaptation to new data and to complex nonlinear patterns. The limitation becomes clear when high-dimensional audio signals are processed.

To address this issue, a dynamic and self-organized feature engineering model was developed. The design aims to reduce the gap between classical feature engineering and deep learning. The structure is parametric and includes established feature selectors [25–27], shallow classifiers [28,29], self-organized information fusion [30]. The first is ATT-BP, which decomposes raw audio signals and forms multiple

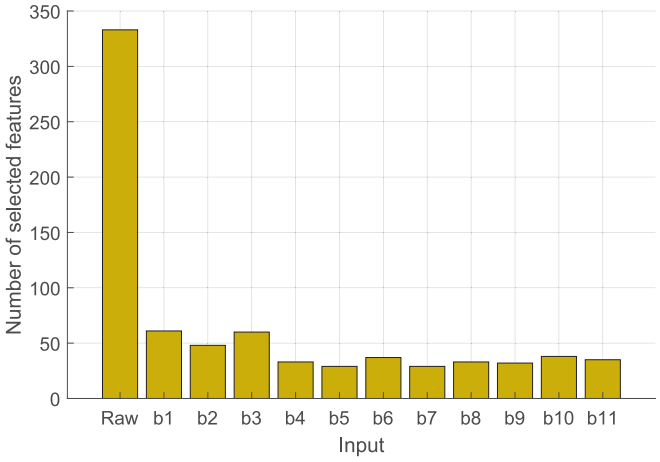

Fig. 8. Selected features by data input. 333 and 435 selected features were extracted from raw sound input and compressed bands (b1 to b11), respectively.

compression bands for further extraction. The second is DBNPat and statistical-based feature extraction and this feature extraction supports hybrid feature extraction and allows dynamic pattern selection for textural (nonlinear) and statistical features [34]. Moreover, multiple feature selectors and classifiers are utilized to use benefits of these models together and IMV provides it. Our DBNPat-based feature engineering model achieved high classification performances on the curated dataset and this dataset contains over 10,000 environmental sound for emotion recognition.

5.1. Comparison with previous studies

SCED is a new-generation research area and remains less established than fields such as ESC and SER. Therefore, direct benchmarks with the same dataset and the same class structure are not available. A fair evaluation of the ATT-BP and DBNPat model still requires comparison with strong methods from related audio classification tasks.

Most existing studies (ESC and SER models) depend on standard datasets for instance ESC-50, UrbanSound8K or RAVDESS. These datasets contain relatively few samples, and many reported models employ computationally intensive deep learning architectures. Comparative results are presented in Table 4.

Table 4 showcases that the DBNPat-based model reached an

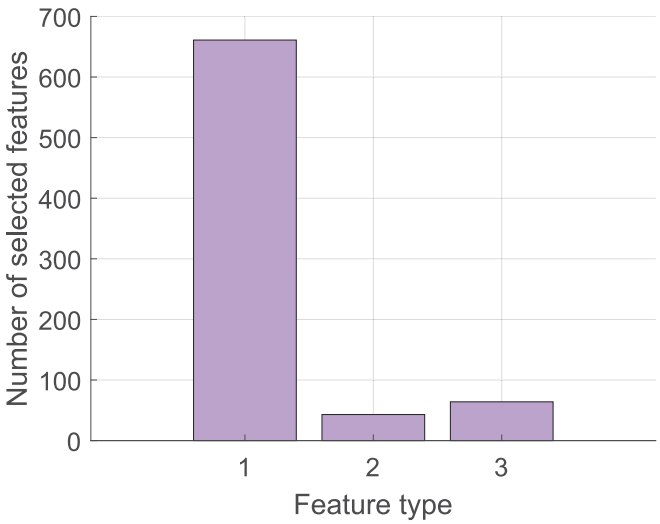

Fig. 9. Selected features by feature type. Among 768 (=256 × 3) selected features, 661, 43, and 64 were DBNPat-generated textural (nonlinear) features (1), statistical features (2), and statistics of DBNPat-generated textural (nonlinear) features (3), respectively.

accuracy of 87.28 %. This result is higher than the accuracy values reported in several deep learning studies. Piczak [21] reported 64.50 % accuracy, and Salamon and Bello [39] reported 79.00 % accuracy on environmental sound datasets using CNN models. In Speech Emotion Recognition, Issa et al. [40] reached 71.61 % accuracy on the RAVDESS dataset with a deep CNN. The CNNs are powerful but they are more expensive than a feature engineering model.

Our study utilized a substantially larger dataset of 10,017 clips, exceeding the size of common benchmarks like ESC-50 and RAVDESS. Despite the increased dataset size, our approach using an SVM classifier with handcrafted features achieved higher accuracy than the deep learning models while maintaining lower computational complexity.

These comparisons support two main points. First, the ATT-BP and DBNPat feature generation methods extracted discriminative high-level patterns from audio with lower computational cost than deep neural networks. Second, the DBNPat-based model remained stable and scalable. High performance was maintained on a large and imbalanced dataset, a common challenge in community emotion detection.

5.2. Computational complexity and runtime analysis

The lightweight DBNPat-based model was designed to operate efficiently on standard hardware. No GPU support or high-cost processing units were required. To support this claim, the theoretical time complexity and the actual runtime performance were evaluated. The DBNPat-based method was then compared with deep learning benchmarks.

Theoretical Complexity:

The proposed method consists of two main phases: feature extraction (ATT-BP and DBNPat) and classification (SVM).

ATT-BP Decomposition: The attention binary pattern processes the signal in a single pass with a sliding window. For a signal of length N, the complexity is linear, denoted as $O(N)$.

DBNPat Feature Generation: This phase involves bitwise operations and histogram extraction on the decomposed signals. Since it iterates through the generated map once, its complexity is also $O(N)$.

Classification (SVM): The prediction complexity of a linear SVM is $O(d)$, where d is the number of features. Since d is fixed after feature selection (Iterative Neighborhood Component Analysis), the classification is extremely fast.

In contrast, deep learning models like VGG16 or ResNet-50 involve millions of floating-point operations (FLOPs) due to convolution and matrix multiplications, typically scaling with $O(K^2.M.N)$ per layer, where K is the kernel size and M, N are feature map dimensions.

Empirical Runtime Analysis:

The average time needed to extract features and classify one four-second audio sample was measured. All tests were carried out on a PC with an Intel Core i7 processor and 32 GB RAM, using MATLAB 2023a. The results are listed in Table 5. The proposed method processed each sample much faster than the pre-trained deep learning models.

Table 5 illustrates that only ~ 0.019 s were needed to process a single sound clip. This time is about 20 times faster than ResNet50 and about 22 times faster than VGG16. The low processing cost indicates that the model can be used in real-time systems and on edge devices with limited hardware resources.

5.3. Key Findings and Advantages

The hybrid extraction structure produced an enriched and multilevel feature set. Nonlinear features and statistical features were included together, and this combination provided strong emotion-related cues. The nonlinear features generated by DBNPat were the dominant group. A total of 86.1 % of the selected features came from DBNPat, and all of them were textural features (Fig. 9). This result showed that nonlinear extraction played an important role in SCED.

The three DBNPat kernels were examined in separate groups. The lower ternary kernel produced the highest number of useful features. The signum kernel followed with a close value. Both kernels extracted different but complementary patterns from the signals. The ATT-BP compression bands also contributed more than half of the selected features (Fig. 8). This observation showed that ATT-BP acted as an effective decomposition step and preserved emotion-related information in the audio data.

NCA was detected as the most effective selector, and SVM reached

Table 5
Comparison of the proposed method with state-of-the-art studies in environmental sound and emotion recognition.

Model	Feature Extraction Time (sec)	Classification Time (sec)	Total Time (sec)
VGG16 (Pre-Trained)	0.4250	0.0020	0.4270
ResNet50 (Pre-Trained)	0.3810	0.0025	0.3835
Proposed (ATT-BP + DBNPat)	0.0185	0.0004	0.0189

Table 4
Comparison of the proposed method with state-of-the-art studies in environmental sound and emotion recognition.

Study	Domain/Task	Dataset (Size)	Method	Accuracy (%)	Computational complexity
Piczak [21]	Env. Sound Classification	ESC-50 (2,000 clips)	CNN (Log-Mel Spectrograms)	64.50	Exponential
Salamon and Bello [39]	Env. Sound Classification	UrbanSound8K (8,732 clips)	CNN + Data Augmentation	79.00	Exponential
Issa et al. [40]	Speech Emotion Recognition	RAVDESS (1,440 clips)	Deep CNN + Mel-Spectrogram	71.61	Exponential
Proposed Method	Community Emotion (SCED)	Proposed Dataset (10,017 clips)	ATT-BP + DBNPat + SVM	87.28	Linear

the highest classifier-wise accuracy (Fig. 6). These results indicated that both methods matched the needs of the SCED task. The IMV fusion step combined the classifier outputs and formed a stronger final result. The fused output exceeded the performance of each individual classifier.

Stable performance was observed across all metrics. This stability remained even with a mildly imbalanced dataset. No synthetic balancing methods, including SMOTE or ADASYN, were used, and natural acoustic patterns were not distorted. Even under this condition, the F1-score and geometric mean stayed above 80 %. These outcomes showed that the handcrafted feature set was strong enough to represent minority classes without artificial resampling.

The benefits of this model are:

- The study dataset is the largest SCED dataset to be used in exploring emotions within environmental sounds, regardless of language or specific context.
- The use of the ATT-BP compression function enhances the diversity and richness of the feature set, resulting in higher classification accuracy.
- The hybrid feature extraction approach is able to pick important minute details from the sound data.
- DBNPat is an efficient nonlinear feature extractor and hence yielded high classification performance.
- The information fusion phase effectively consolidated complex data and yielded an accurate final results.
- The proposed model is able to handle effectively a large dataset and obtain high classification performance.

6. Conclusions and future Works

This research introduced a novel feature engineering model for Sound-based Community Emotion Detection (SCED). The model implements a comprehensive machine learning pipeline that combines DBNPat-based feature generation with ATT-BP compression for multilevel feature extraction. The pipeline employs three feature selectors (NCA, Chi2, and mRMR) and two classifiers (kNN and SVM), generating six initial outcomes. These outcomes are further refined through Iterative Majority Voting (IMV), which produces four additional voted outcomes. The final classification result is selected through a greedy algorithm that identifies the outcome with maximum performance, exemplifying the model's self-organized nature.

To validate our approach, we curated a new SCED dataset containing over 10,000 audio samples. The DBNPat-based model achieved 87.28 % classification accuracy on this dataset, with all performance metrics (UAR, UAP, F1-score, and geometric mean) exceeding 80 %. These results demonstrate the effectiveness of our feature engineering approach for emotion detection in environmental sounds.

While the results are promising, several limitations should be acknowledged. The model has been validated only on a mildly imbalanced dataset, and its performance on perfectly balanced data remains to be investigated. The three-second audio samples, while practical, may limit the capture of extended emotional patterns and temporal information. Additionally, the current three-class emotion categorization may not fully represent the nuanced spectrum of emotions present in environmental scenes.

Future research directions include exploring additional deep learning architectures as templates for feature extraction rules and investigating alternative pattern selection functions to enhance the self-organization process. The emotion classification framework could be expanded beyond the current three classes to enable more refined emotional discrimination. Furthermore, the methodology shows potential for adaptation to other domains such as environmental monitoring, security systems, and healthcare applications where sound analysis provides valuable insights.

CRediT authorship contribution statement

Arif Metehan Yildiz: Methodology, Conceptualization, Software, Writing – original draft. **Prabal Datta Barua:** Investigation, Visualization. **Mehmet Baygin:** Data curation, Formal analysis, Validation. **Sengul Dogan:** Writing – original draft, Data curation, Formal analysis, Validation. **Turker Tuncer:** Supervision, Methodology, Software. **Massimo Salvi:** Visualization, Writing – review & editing. **Ru-San Tan:** Writing – review & editing, Validation. **U.R. Acharya:** Writing – review & editing, Supervision, Conceptualization.

Funding

This research received no external funding.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

[1] M. Gallagher, Sound as affect: Difference, power and spatiality, Emot. Space Soc. 20 (2016) 42–48.
[2] M. Soleymani, M. Pantic, T. Pun, Multimodal emotion recognition in response to videos, IEEE Trans. Affect. Comput. 3 (2011) 211–223.
[3] S. Albanie, A. Nagrani, A. Vedaldi, A. Zisserman, Emotion recognition in speech using cross-modal transfer in the wild, in: Proceedings of the 26th ACM International Conference on Multimedia, 2018, pp. 292–301.
[4] A. Kanavos, I. Perikos, I. Hatzilygeroudis, A. Tsakalidis, Emotional community detection in social networks, Comput. Electr. Eng. 65 (2018) 449–460.
[5] C. Clavel, Z. Callejas, Sentiment analysis: from opinion mining to human-agent interaction, IEEE Trans. Affect. Comput. 7 (2015) 74–93.
[6] R. Cowie, E. Douglas-Cowie, N. Tsapatsoulis, G. Votsis, S. Kollias, W. Fellenz, J. G. Taylor, Emotion recognition in human-computer interaction, IEEE Signal Process Mag. 18 (2001) 32–80.
[7] W. Bucci, Pathways of emotional communication, Psychoanal. Inq. 21 (2001) 40–70.
[8] E. Cannoni, G. Pinto, A.S. Bombi, Typical emotional expression in children's drawings of the human face, Curr. Psychol. 42 (2023) 2762–2768.
[9] A.C. Frenzel, T. Goetz, K. Stockinger, Emotions and emotion regulation, Routledge, Handbook of educational psychology, 2024, pp. 219–244.
[10] J.F. Courtney, Decision making and knowledge management in inquiring organizations: toward a new decision-making paradigm for DSS, Decis. Support Syst. 31 (2001) 17–38.
[11] C. Peter, B. Urban, Emotion in human-computer interaction, Expanding the Frontiers of Visual Analytics and Visualization (2012) 239–262.
[12] S.K. Khare, V. Blanes-Vidal, E.S. Nadimi, U.R. Acharya, Emotion recognition and artificial intelligence: a systematic review (2014–2023) and research recommendations, Inf. Fusion 102019 (2023).
[13] M. Jafari, A. Shoeibi, M. Khodatars, S. Bagherzadeh, A. Shalbaf, D.L. García, J. M. Gorriz, U.R. Acharya, Emotion recognition in EEG signals using deep learning methods: a review, Comput. Biol. Med. 107450 (2023).
[14] Z. He, Z. Li, F. Yang, L. Wang, J. Li, C. Zhou, J. Pan, Advances in multimodal emotion recognition based on brain–computer interfaces, Brain Sci. 10 (2020) 687.
[15] M. Maithri, U. Raghavendra, A. Gudigar, J. Samanth, P.D. Barua, M. Murugappan, Y. Chakole, U.R. Acharya, Automated emotion recognition: current trends and future perspectives, Comput. Methods Programs Biomed. 215 (2022) 106646.
[16] M. Egger, M. Ley, S. Hanke, Emotion recognition from physiological signal analysis: a review, Electron. Notes Theor. Comput. Sci. 343 (2019) 35–55.
[17] M.S. Hossain, G. Muhammad, An emotion recognition system for mobile applications, IEEE Access 5 (2017) 2281–2287.
[18] A.M. Yildiz, M. Tanabe, M. Kobayashi, I. Tuncer, P.D. Barua, S. Dogan, T. Tuncer, R.-S. Tan, U.R. Acharya, FF-BTP Model for Novel Sound-based Community Emotion Detection, IEEE Access (2023).
[19] P.B. Dasgupta, Detection and analysis of human emotions through voice and speech pattern processing, arXiv preprint arXiv:1710.10198, (2017).
[20] F. Weninger, F. Eyben, B.W. Schuller, M. Mortillaro, K.R. Scherer, On the acoustics of emotion in audio: what speech, music, and sound have in common, Front. Psychol. 4 (2013) 292.
[21] K.J. Piczak, ESC: Dataset for environmental sound classification, pp. 1015-1018.
[22] J. Salamon, C. Jacoby, J.P. Bello, A dataset and taxonomy for urban sound research, pp. 1041-1044.

[23] S.R. Livingstone, F.A. Russo, The Ryerson Audio-Visual Database of Emotional Speech and song (RAVDESS): a dynamic, multimodal set of facial and vocal expressions in north American English, PLoS One 13 (2018) e0196391.
[24] F. Burkhardt, A. Paeschke, M. Rolfes, W.F. Sendlmeier, B. Weiss, A database of german emotional speech, pp. 1517-1520.
[25] J. Goldberger, G.E. Hinton, S. Roweis, R.R. Salakhutdinov, Neighbourhood components analysis, Adv. Neural Inf. Proces. Syst. 17 (2004) 513–520.
[26] H. Liu, R. Setiono, Chi2: Feature selection and discretization of numeric attributes, Proceedings of 7th IEEE international conference on tools with artificial intelligence, IEEE, 1995, pp. 388-391.
[27] H. Peng, F. Long, C. Ding, Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy, IEEE Trans. Pattern Anal. Mach. Intell. 27 (2005) 1226–1238.
[28] L.E. Peterson, K-nearest neighbor, Scholarpedia 4 (2009) 1883.
[29] W.S. Noble, What is a support vector machine? Nat. Biotechnol. 24 (2006) 1565–1567.
[30] A. Dogan, M. Akay, P.D. Barua, M. Baygin, S. Dogan, T. Tuncer, A.H. Dogru, U. R. Acharya, PrimePatNet87: Prime pattern and tunable q-factor wavelet transform techniques for automated accurate EEG emotion recognition, Comput. Biol. Med. 138 (2021) 104867.
[31] A. Vince, A framework for the greedy algorithm, Discret. Appl. Math. 121 (2002) 247–260.
[32] T. Ojala, M. Pietikainen, T. Maenpaa, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, IEEE Trans. Pattern Anal. Mach. Intell. 24 (2002) 971–987.
[33] F. Bieder, R. Sandkühler, P.C. Cattin, Comparison of methods generalizing max-and average-pooling, arXiv preprint arXiv:2103.01746, (2021).
[34] F. Kuncan, K. Yılmaz, M. Kuncan, Sensör işaretlerinden cinsiyet tanıma için yerel ikili örüntüler tabanlı yeni yaklaşımlar, Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi, 34 2173-2186.
[35] G.E. Hinton, Deep belief networks, Scholarpedia 4 (2009) 5947.
[36] D.M.W. Powers, Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation, arXiv preprint arXiv:2010.16061, (2020).
[37] S. Dargan, M. Kumar, M.R. Ayyagari, G. Kumar, A survey of deep learning and its applications: a new paradigm to machine learning, Arch. Comput. Meth. Eng. 27 (2020) 1071–1092.
[38] M.A. Ganaie, M. Hu, A.K. Malik, M. Tanveer, P.N. Suganthan, Ensemble deep learning: a review, Eng. Appl. Artif. Intel. 115 (2022) 105151.
[39] J. Salamon, J.P. Bello, Deep convolutional neural networks and data augmentation for environmental sound classification, IEEE Signal Process Lett. 24 (2017) 279–283.
[40] D. Issa, M.F. Demirci, A. Yazici, Speech emotion recognition with deep convolutional neural networks, Biomed. Signal Process. Control 59 (2020) 101894.