DOI: 10.61782/fa.2023.1309

forumacusticum 2023

MACHINE LEARNING FRAMEWORK FOR THE ACOUSTIC DETECTION OF THE QUEEN BEE PRESENCE

Luca Barbisan
Dept. of Electronics and Telecommunications
Politecnico di Torino, Torino, 10129 Italy
luca.barbisan@polito.it

Fabrizio Riente
Dept. of Electronics and Telecommunications
Politecnico di Torino, Torino, 10129 Italy
fabrizio.riente@polito.it

ABSTRACT

Honeybees are one of the most important pollinators in nature for both crop production and biodiversity preservation. The increase in bee mortality observed in the last decade motivated the development of continuous monitoring systems to better understand this phenomenon. Different solutions have been presented in the literature, and particularly sound analysis appears the most promising among the non-invasive techniques. In this context, we developed a machine learning framework for the analysis of the sound produced by bees for the detection of the queen bee's presence. The presence of the queen is an important indicator of the colony's health. In this work, we investigated Short Time Fourier Transform and Mel Frequency Cepstral Coefficient audio features with support vector machines and neural network classifiers. The results indicate the potential of machine learning methods for supporting the researchers' study and beekeepers in managing such important insects.

Keywords: *sound classification, machine learning, queen bee detection, bee monitoring.*

1. INTRODUCTION

The importance of honey bees (Apis mellifera L.) is not limited to the production of honey, beeswax, royal jelly, and propolis, but they provide pollination service for about 70% of the crop in the world. In the last decade, many stress factors such as climate change, the use of pesticides, and intensive agriculture have led to a decline in the honeybee colonies [1]. This situation emphasized the necessity of continuous monitoring to investigate the motivation for such a decline in the bee colonies and support both researchers and beekeepers. In the literature, several techniques have been proposed to tackle this problem. Some solutions collect environmental parameters inside the beehive [2–4]. Others, exploit computer vision to track bees' movements [5, 6]. In some cases, the proposed solutions require the modification of the beehive for the installation of sensors, which is impractical for a real device. In recent years, non-invasive techniques, based on audio processing, are becoming interesting, thanks also to the miniaturization of sensing technologies. Honeybees communicate using a combination of vibroacoustic signals [7, 8]. The sound level amplitude and frequency depend on the activity of the colony. Frequencies can reach up to 3000 Hz in case of defensive reaction [9] [10]. Researchers have proved the correlation between signal amplitudes and frequencies of honeybees and some events such as swarming [9, 11] and queen presence [12, 13]. In this scenario, the representation of acoustic signals, combined with machine learning methods can lead to the development of automatic systems that can discriminate among different events that can characterize the colony state. Authors in [12–14], explored Short Time Fourier Transform (STFT) and Mel Frequency Cepstral Coefficient (MFCC) audio features with a combination of classifiers. Particularly, they investigated support vector machines (SVMs) and convolutional neural networks (CNNs), and neural network (NNs) classifiers.

In this work, we developed a machine learning framework for the detection of the queen bee presence using only audio features (MFCC and STFT). We analyzed the performance of both SVM and NN classifiers and compared the results with [13]. The paper is organized as follows. Section 2 presents the developed framework, introducing the adopted dataset, feature extraction and classification methods. Section 3 reports the experiments performed and summarizes the obtained results. Finally, section 4 conclude the paper giving insight on future research directions.

2. DEVELOPED FRAMEWORK

In this work, we developed a framework as a Python library, that can be used for the investigation of machine learning models for audio classification tasks. It makes it possible to create a list of experiments and to compare the results of different classification algorithms applied to a selection of audio features and extensible to any dataset that contains audio files. Each experiment performs steps that are schematically represented in figure 1. Dataset audio files are first split into chunks for later processing. From there, audio features are extracted and split among training and test data. The model is trained, performing the k-fold cross-validation and finally, the final evaluation is executed extracting a log file with all the metrics. These steps can be iterated in different experiments in order to compare the effects of different feature parameters, features, and classificator hyperparameters.

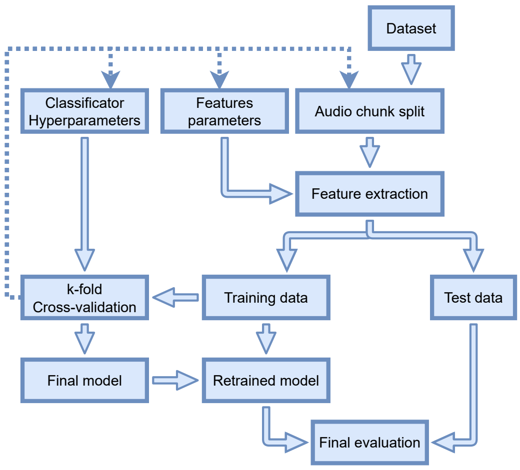

Figure 1: Schematic representation of the framework pipeline.

2.1 Dataset and split into audio chunks

The framework has been designed to allow the use of a wide range of datasets. It offers the flexibility to integrate and work with any dataset that contains audio data creating new custom classes. The dataset is not limited to only audio data, it can contain also other information related to the audio samples that can be useful to be integrated as a feature when performing the classification. As an example, the queen bee detection can be improved by combining the atmospheric conditions that are usually available in the datasets as a CSV file.
The framework executes a preprocessing step that consists of splitting the audio files into audio chunks with the same length and resampling to the same frequency, this step facilitates the analysis and extraction of consistent audio features and ensures uniformity and compatibility across different datasets that can contain audio files recorded with different sample rates and different duration.

2.2 Feature extraction

The feature extraction step currently supports two commonly used techniques for audio applications: Mel Frequency Cepstral Coefficients and Short Time Fourier Transform. These can be used singularly or combined together, however, the framework is designed to be expandable allowing researchers to create custom feature extraction classes to extract new features.

MFCC: is a widely used technique in audio signal processing and feature extraction. It captures the characteristics of and audio signal by representing its power spectrum on Mel-frequency scale and then applying a discrete cosine transform to obtain the n cepstral coefficients. It provides a compact representation of the spectral components averaged over the entire duration of the chunk.

STFT: is another fundamental technique for analyzing audio signals. It considers the signal split into partially overlapped windows, compute the Fourier transform, and then calculates the magnitude of the power spectrum for each window properly filtered with a 'Hann' windowing function. Also, this feature is averaged over the entire duration of the chunk in order to generate a compact representation of the audio file.

2.3 Training test split and cross-validation

The extracted features are randomly split into two subsets: the training data and the test data. This subdivision can be configured specifying the percentage of the initial dataset that will be used to do the final test. To further evaluate the model capability to generalize to unseen data identifying potential issues such as overfitting or underfitting, the framework incorporates the k-fold technique. The training data is divided into k equally sized subsets, then the model is trained k times using each time different k-1 folds and evaluated on the remaining one. The statistics are computed averaging the results of the k iterations, ensuring the development of robust and generalized audio classification models.

2.4 Final model training and evaluation

The framework proceeds to train the final model using the entire training set, this step ensures that the model is trained on the maximum amount of available data potentially improving its performance and ability to generalize. The trained model is evaluated using the test set which serves as an independent validation set, this helps to confirm the statistical metrics computed during the cross-validation process and produce a final model that can be used for the deployment of the application.

3. RESULTS

In order to evaluate our framework, we tested it using the NU-Hive [15] project that contains audio acquired in honey bee hives and it is used in other works such as [13] [16]. It contains 573 audio files of 10 minutes duration each with a sampling rate of 32kHz.
We configured the framework to generate chunks of 1 second, without overlapping them, this produced 336502 chunks. Then, we extracted the MFCC features using $n = 20$ and subdivided the obtained dataset into training data (95%) and test data (5%) in agreement with [13]. The training data is then used with a 10 fold cross validation to compare two different classifiers: i) SVM with $C = 1$ and $kernel = RBF(RadialBasisFunction)$ that replicates the SVM classifier configuration that obtained the best results with the random split setup in [13]; ii) NN with an architecture based on 20 inputs, hidden layers=256 LeakyReLU, 32 LeakyReLU is a similar architecture to the CNN used in [13] were the convolutional layers have been removed. From the cross validation results we obtain the values shown in Table 1(a), here we reported the mean value and the standard deviation for the metrics computed at the end of the 10-fold iterations. Figure 2(a) displays the training and cross-validation curves for both loss and accuracy of the neural network. It is evident that increasing the number of epochs leads to an overfitting problem. In Figure 2(b), the graph illustrates the performance of the SVM with varying gamma values. Lower gamma values result in underfitting, while higher gamma values cause a clear separation between the training and cross-validation curves, indicating the presence of overfitting. They are similar to the results obtained in [13] with the SVM (AUC score ~ 0.91) and CNN (AUC score ~ 0.99) in the random split test. The numerical value of the CNN AUC score reported in [13] has been derived from a figure, being missing in the text.

Table 1: (a) Cross-validation results and (b) final evaluation results for both SVM and NN classifiers.

(a)

algorithm	accuracy	f1-score	ROC AUC
SVM	0.9879±4.49e-04	0.9879±4.86e-04	0.9991±2.92e-05
NN	0.9889±10.54e-04	0.9890±11.04e-04	0.9993±10.12e-05

(b)

algorithm	accuracy	recall	precision	f1-score
SVM	0.9886	0.9902	0.9873	0.9888
NN	0.9885	0.9849	0.9923	0.9886

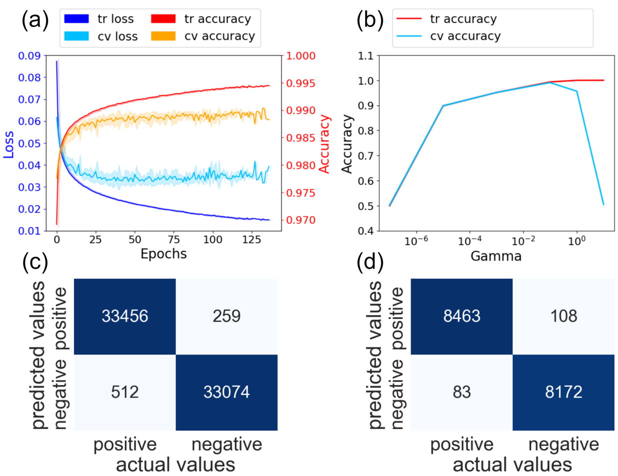

Figure 2: Training (tr) and cross-validation (cv) scores of (a) NN and (b) SVM, final evaluation confusion matrix of (c) NN and (d) SVM.

Finally, for both the classification algorithms we trained a final model and the results are reported in the confusion matrix (Figure 2(c)-(d)) and the derived metrics (Table 1(b)) in order to check the previously obtained results. Unfortunately, in [13] the authors did not report the test scores of the models. However, the obtained metrics show a good behavior of the models with unseen data. In particular, an interesting result is obtained with the neural network that achieved a high f1-score both in cross-validation (=0.9889) and in the final evaluation (=0.9886) considering that the generated model is significantly smaller compared to the CNN tested in [13] that contains additional four convolutional layers, this has to be taken into account when the target device that will run the model has a limited amount of resources like in this case where small devices are used for bee monitoring.

4. CONCLUSION

In conclusion, this work demonstrates the effectiveness of the developed framework as a valuable tool for audio classification problems, specifically in the task of queen bee presence detection comparing the results obtained by the two algorithms SVM and NN adopted in [13]. Future developments will focus on implementing advanced feature extraction method investigating other machine learning algorithms and explore additional parameters contributing to the advancement of machine learning in audio classification. The framework's modular design and ability to incorporate various features and classification algorithms can play an important role in identifying the most suitable approaches for queen bee presence and other classification tasks.

5. REFERENCES

[1] J.-P. Faucon, L. Mathieu, M. Ribiere, A.-C. Martel, P. Drajnudel, S. Zeggane, C. Aurieres, and M. F. A. Aubert, "Honey bee winter mortality in france in 1999 and 2000," *Bee World*, vol. 83, no. 1, pp. 14–23, 2002.

[2] C. Stefania, S. Spinsante, T. Alessandro, and O. Simone, "A smart sensor-based measurement system for advanced bee hive monitoring," *Sensors*, vol. 20, no. 9, 2020.

[3] A. Rafael Braga, D. G. Gomes, R. Rogers, E. E. Hassler, B. M. Freitas, and J. A. Cazier, "A method for mining combined data from in-hive sensors, weather and apiary inspections to forecast the health status of honey bee colonies," *Computers and Electronics in Agriculture*, vol. 169, p. 105161, 2020.

[4] F. Bellino, G. Turvani, U. Garlando, and F. Riente, "An integrated multi-sensor system for remote bee health monitoring," in *2022 IEEE Workshop on Metrology for Agriculture and Forestry (MetroAgriFor)*, pp. 334–338, 2022.

[5] T. N. Ngo, K.-C. Wu, E.-C. Yang, and T.-T. Lin, "A real-time imaging system for multiple honey bee tracking and activity monitoring," *Computers and Electronics in Agriculture*, vol. 163, p. 104841, 2019.

[6] C. Yang and J. Collins, "A model for honey bee tracking on 2d video," in *2015 International Conference on Image and Vision Computing New Zealand (IVCNZ)*, pp. 1–6, 2015.

[7] M. Drosopoulos, S. Claridge, *Insect Sounds and Communication: Physiology, Behaviour, Ecology, and Evolution (1st ed.)*. CRC Press, 2005.

[8] F. Hunt, J.H. Richard, "Intracolony vibroacoustic communication in social insects," *Insect. Soc*, vol. 60, pp. 403–417, 2013.

[9] A. Žgank, "Acoustic monitoring and classification of bee swarm activity using mfcc feature extraction and hmm acoustic modeling," in *2018 ELEKTRO*, pp. 1–4, 2018.

[10] W. Kirchner, "Acoustical communication in honeybees," *Apidologie*, vol. 24, no. 3, pp. 297–307, 1993.

[11] H. Eren, L. Whiffler, and R. Manning, "Electronic sensing and identification of queen bees in honeybee colonies," in *IEEE IMTC Proceedings*, vol. 2, pp. 1052–1055 vol.2, 1997.

[12] A. Terenzi, S. Cecchi, and S. Spinsante, "On the importance of the sound emitted by honey bee hives," *Veterinary Sciences*, vol. 7, no. 4, 2020.

[13] I. Nolasco, A. Terenzi, S. Cecchi, S. Orcioni, H. L. Bear, and E. Benetos, "Audio-based identification of beehive states," in *ICASSP 2019-2019 IEEE ICASSP*, pp. 8256–8260, IEEE, 2019.

[14] A. Terenzi, N. Ortolani, I. Nolasco, E. Benetos, and S. Cecchi, "Comparison of feature extraction methods for sound-based classification of honey bee activity," *IEEE/ACM TASLP*, vol. 30, pp. 112–122, 2022.

[15] S. Cecchi, A. Terenzi, S. Orcioni, P. Riolo, S. Ruschioni, and N. Isidoro, "A preliminary study of sounds emitted by honey bees in a beehive," in *Audio Engineering Society Convention 144*, Audio Engineering Society, 2018.

[16] A. Terenzi, N. Ortolani, I. Nolasco, E. Benetos, and S. Cecchi, "Comparison of feature extraction methods for sound-based classification of honey bee activity," *IEEE/ACM TASLP*, vol. 30, pp. 112–122, 2021.