Intent Detection in Speech Recognition Based on Machine Learning and Deep Learning

Zahra Karimi
Politecnico di Torino
s302612
Torino, Italy
s302612@studenti.polito.it

Ali Abbasi
Politecnico di Torino
s290007
Torino, Italy
s290007@studenti.polito.it

***Abstract*—Intent detection is one of the important topics in the fields of natural language processing (NLP) and speech recognition. In this study, a one-dimensional convolutional neural network called 1D-CNN-Intent is developed to detect the intent of audio files. Moreover, several base machine learning algorithms on MFCC and spectrogram features of audio files were evaluated. Experimental results show that deep learning approaches (1D-CNN-Intent) successfully extract features for Intent classification (classify up to 87% for accuracy).**

I. PROBLEM OVERVIEW

Intent detection in audio files is a classification problem in machine learning that can be very useful in developing audio assistants and audio-based applications. In this kind of problem, audio files are taken from users, and then the class of intention is recognized by machine learning methods. Here is the problem of detecting intent in the audio files dataset, which is categorized into 7 classes. The dataset consists of a collection of audio files in a WAV format, and each record is characterized by several attributes. The dataset is divided into two parts:

- *Development dataset:* it contains 9854 records in the form of a CSV file, which includes various features. The most important features are the path of the audio file (audio files in the root folder), the type of action required through the intent (Action), and the device involved by intent (Object).
- *Evaluation dataset:* it contains 1455 records in the form of a CSV file, and its features are the same as the development datasets, only with the difference that we do not have access to its labels. Therefore, the proposed model and machine learning algorithms should classify the intent of each record.

In this study, the final intention label can be calculated using action and object attributes. Therefore, the values of these two attributes are combined and create a new label for each record. In the development data set, this combination has created 7 new classes. According to 9854 available records in the development dataset, the number of samples for each class is shown in Table I. According to Table I, the maximum number of samples belong to the "IncreaseVolume" class with 2614 samples. The least number of samples are for "DeactivateLights" and "ActivateMusic" with 552 and 791 samples, respectively. For a better understanding, the waveform of a random sample audio signal of each class is shown in Figure 1.

Class Name	Class Encode	Number of Sample
ActivateMusic	0	791
ChangeLanguagenone	1	1113
DeactivateLights	2	552
DecreaseHeat	3	1189
DecreaseVolume	4	2386
IncreaseHeat	5	1209
IncreaseVolume	6	2614

TABLE I
THE NAME AND NUMBER OF CLASSES IN THE AUDIO FILES DATASET

The plot Wav sample for each class is shown in Figure 1, where there is a problem of silence at the beginning and end of each audio file which can be solved by trimming. In addition, the length of the signals based on the distribution of each class after the trimming operation is shown in Figure 2.

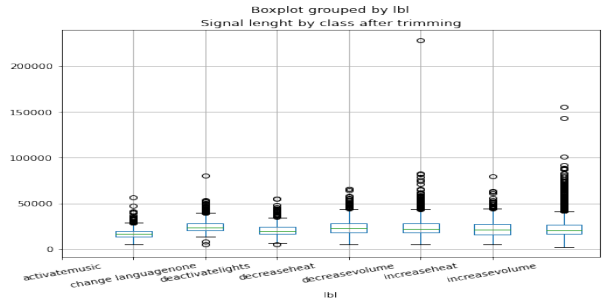

Fig.2: Distribution of samples for each class

In this dataset, all signals have been sampled at the same rate. but according to Figure 2, The duration of each audio record is different. On the other hand, some samples have very different lengths compared to the average length of all records. However, machine learning methods and deep methods require the same length of features. Therefore, for both of these

methods, a solution should be provided for feature extraction with a fixed length. However, there are some points about the preparation of audio signals(fixed length, normalization, split, etc.) that are further mentioned in section II.

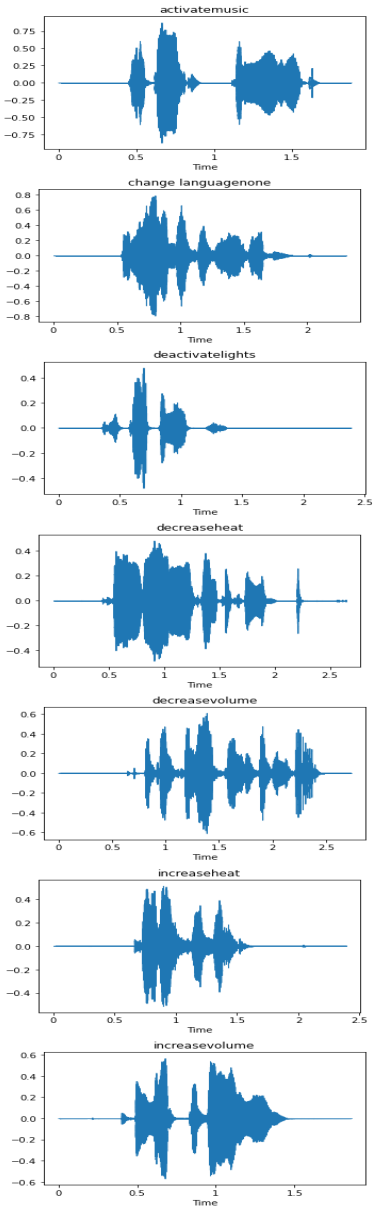

Fig.1: lot of a randomly selected sample of every intent class, (a) ActivateMusic, (b) ChangeLanguagenone, (c) DeactivateLights, (d) DecreaseHeat, (e) DecreaseVolume, (f) IncreaseHeat, and (g) IncreaseVolume

II. PROPOSED APPROACH

Deep learning is the new science of machine learning, in which neural networks are trained with more layers, different types of layers, and many hyperparameters. Usually, these neural networks are composed of varying levels of nonlinear operators. Deep-based methods have performed better than other methods, especially machine learning methods in speech recognition[1]. In this section, the new 1D-CNN-Intent approach for classification intent in audio files is described.

A. Data preprocessing

Speech signal preprocessing is very important in the process of learning and classifying speech intention by deep learning and machine learning methods. Therefore, we will try to examine the main stages of pre-processing of the speech signal that was carried out in this study.

- Labelencoder: according to Table I and with the help of the LabelEncoder method, all class names are converted into a numeric type.
- Trimming silence: silence has no relevant information, so it can remove from the start and the end of each audio file and makes the algorithms focus on the parts of the audio signal that contain more information. The audio file length after trimming silence is shown in Figure 3.

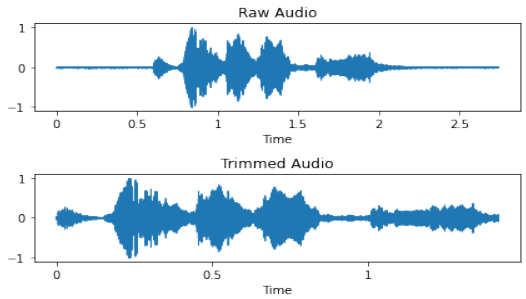

Fig.3: Trimming silence audio

- *Equalization of Scale:* machine learning algorithms and deep algorithms require the same length of features; therefore, the length of all audio signals should be the same. All the audio signals whose length is more than 45 thousand are reduced to 45, and also those whose length is less than zero are filled with zero padding at the end.
- *Normalization:* data normalization is one of the crucial techniques of data preprocessing before the training stage of different methods[2]. It changes the value of the features to a new scale, which increases the accuracy of the machine-learning algorithms[3]. In this study, the Standard Scaler method has been used, which is formulated as follows.

$$X' = \frac{X - X_{Mean}}{X_{var}} \tag{1}$$

- *Feature Extraction:* here the goal is to extract features from prepared signals for machine learning algorithms based on two methods of MFCC and spectrogram, although, the 1D-CNN-Intent approach does not need them. As a result, approximately 768 features are extracted based on the average, maximum, and minimum rows values.

B. Model selection

After pre-processing and data preparation, several machine learning algorithms and a deep learning model(1D-CNN-Intent

approach) are selected in this section as the intention classifier model in audio files.

- *1D-CNN-Intent:* The convolutional neural network is of more interest in this study because it can automatically extract new features from the entire audio signal data. The proposed method receives pre-processed audio signal data directly and extracts features with the help of convolutional layers. Finally, it classifies audio files with Fully connected layers. Considering the length of the audio signals, which here is a fixed length of 45,000, the proposed convolutional layers are set with large kernels such as 500. A large kernel size makes it possible to extract local and important features from audio signals. The proposed model includes several dropout layers and batch normalization layers to decrease overfitting.
- *Support vector machines (SVM):* Support vector machine is a supervised machine learning algorithm.
- *Random Forest(RF)*: Random Forest is a supervised machine learning algorithm.
- *Decision Tree (DT):* Decision Tree is a supervised machine learning algorithm.

C. Hyperparameters tuning

Machine learning algorithms have some parameters that can play an important role in the output of audio signal intent classification. Various methods have been presented to optimize or select the best parameters. In this study, we used the GridSearchCV which is based on machine learning methods. Although the 1D-CNN-Intent approach has many hyperparameters, which are time-consuming and difficult to adjust by methods. Therefore, a manual trial and error method has been used. In Table II, the values of hyperparameters and the best value among them for all methods are shown.

Models	Hyperparameters	Values	Best selected
SVM	kernels	rbf,poly	rbf
	C	0.01,0.1,10,15,100	10
DT	criterion	entropy,gini	entropy
	min-samples-split	2,4,8,16,32	32
RF	criterion	entropy,gini	entropy
	n-estimators	20,40,80,160,200,300	200
1D-CNN-Intent	batch Size	64,128,512,1024	512
	optimizer	Adam,SGD	Adam
	learning-rate	0.0001 ,0.001,0.1,0.5	0.0001
	epochs	100,200,400,500,700	500

TABLE II
MODELS AND HYPERPARAMETERS

According to Table II, a number of values have been considered for each parameter, and eventually, the best value has been selected according to the F1 score. For example, [0.01,0.1,10,15,100] values are considered for the C parameter of the SVM algorithm, and finally, the value of 10 is set for this parameter. In addition, for the 1D-CNN-Intent, the hyperparameters are selected manually after several attempts based on the F1 score.

III. RESULTS

According to the GridSearchCV method, for the SVM algorithm, the C and kernel parameters are set to 10 and 'rbf', respectively. In addition, for the DT algorithm, the criterion parameter is equal to 'entropy', and also 'min-samples-split' parameter is set to 32. For the RF as the last one, the 'entropy' value is set for the criterion parameter, and the value of 200 is set for the n-estimators. All the hyperparameters of the Intent-CNN-1D model are set manually(Table II).
The proposed Intent-CNN-1D model and all machine learning algorithms are evaluated to detect the intent of the audio files dataset. To compare these methods, 80% of the development dataset is used as training data and 20% of that as validation data, and finally, the evaluation dataset is considered as the final test. In this study, we used standard metrics in machine learning evaluation: accuracy, recall, precision, and the F1 score, which are mentioned below.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

$$Precision = \frac{TP}{TP + FP} \tag{3}$$

$$Recall = \frac{TP}{TP + FN} \tag{4}$$

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall} \tag{5}$$

The algorithms are implemented on the training and validation dataset based on the chosen hyperparameters, and the results for different criteria are given in Table III.
The results of Table III show that the proposed 1D-CNN-Intent model has achieved high accuracy with a value of 77% for the F1 score in the test data set and 87% in the evaluation data set. However, this test shows that among the machine learning methods, the SVM algorithm with a value of 56% has performed better for the F1 score. In addition, in Figure 4, the accuracy plot of the training and validation dataset of the proposed 1D-CNN-Intent model is shown in 500 iterations.

Algorithm	Accuracy	precision	recall	F1	evaluation
SVM	0.56	0.54	0.58	0.56	0.48
DT	0.32	0.32	0.32	0.32	-
RF	0.47	0.41	0.57	0.44	-
1D-CNN-Intent	0.77	0.76	0.78	0.77	0.87

TABLE III
COMPARISON OF MODELS WITH DIFFERENT CRITERIA ON VALIDATION AND EVALUATION DATASET

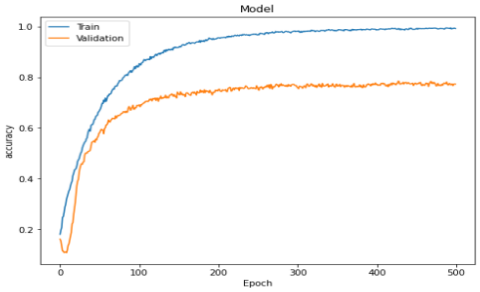

Fig.4: Training and Validation accuracy for the 1D-NN1D-CNN-Intent model

The results of this plot proved that the proposed 1D-CNN-Intent model could detect intent to a high degree. However, it can achieve better detection with necessary changes and improvements in the future.

IV. DISCUSSION

In this study, several machine learning algorithms and a deep method based on convolutional networks were evaluated for detecting intent in audio files. The proposed model 1D-CNN-Intent in the evaluation dataset with a value of 0.87 for the F1 score has proven a high performance compared to other models. The SVM algorithm also achieved a value of 56% for the F1 score. However, there are some cases that can increase the performance of the proposed 1D-CNN-Intent model and machine learning algorithms.

- In order to improve the performance of the machine learning algorithm in detecting intent, first feature extraction is done by the proposed 1D-CNN-Intent model and then feature maps of the last layer are given to these algorithms. This indeed transfers the new features extracted by the proposed model to machine learning models.
- Although the proposed model performed well, to increase its performance, changes can be the type of layers, kernel size, and other hyperparameters optimization.
- In this study, feature extraction for machine learning methods was done based on the spectrogram and MFCC methods. However, extracting features with other methods can also be one of the ways to hence the performance of machine learning algorithms because the performance of these algorithms strongly depends on the input feature.

REFERENCES

[1] Nassif, A.B., et al., Speech recognition using deep neural networks: A systematic review. IEEE access, 2019. 7: p. 19143-19165.

[2] Raju, V.G., et al. Study the influence of normalization/transformation process on the accuracy of supervised classification. in 2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT). 2020. IEEE.

[3] Balaha, H.M., et al., A multi-variate heart disease optimization and recognition framework. Neural Computing and Applications, 2022. 34(18): p. 15907-15944.

[4] Matsane, L., A. Jadhav, and R. Ajoodha. The use of automatic speech recognition in education for identifying attitudes of the speakers. in 2020 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE). 2020. IEEE.

[5] Cortes, C., Vapnik, V.N. (1995). Support-Vector Networks. Machine Learning, 20, 273-297.

[6] Ho, Tin Kam (1995). Random Decision Forests. Proceedings of the 3rd International Conference on Document Analysis and Recognition, Montreal, QC, 14–16 August 1995. pp. 278–282.

[7] Quinlan, J. R. (1986a). Induction of decision trees. Machine Learning, 1(1), 81–106. https://link.springer.com/article/10.1007/BF00116251