Utilisation d'un réseau de neurones pour identifier un bruit dans une séquence audio.

M. Vo C. Mévolhon

Université Paul Sabatier
118 route de Narbonne, 31400 Toulouse

{michael.vo, claire.mevolhon}@univ-tlse3.fr

Résumé

Dans un monde de plus en plus connecté, la reconnaissance automatique des sons environnementaux peut s'avérer être réellement intéressant que ce soit par exemple dans le domaine médical afin d'aider les malentendants mais aussi dans le domaine de la robotique où on essaye de plus en plus de reproduire des facultés humaines. Nous aimerions ainsi automatiquement reconnaître des sons environnementaux mutuellement exclusifs et identifier un bruit particulier dans une séquence audio. Nous avons utilisé ici le corpus ESC-10 spécialement conçu pour classifier 10 sons environnementaux différents. Malheureusement, les réseaux de neurones traitant le son n'existent pas à l'heure actuelle nous choisissons de convertir les séquences audios en spectrogrammes afin d'avoir une représentation spectrale du signal que l'on analysera ainsi comme une image. Nous avons ainsi effectué différentes convolutions suivi alternativement d'une fonction d'activation Tanh ou d'une ReLU. Nous avons ensuite enchaîné avec 1 couche dense Tanh avec du dropout. Pour distinguer 10 sons de manières exclusives : nous avons donc élaboré une couche finale à 10 neurones avec la fonction d'activation softmax.

Nous avons entraîné notre réseau sur un corpus de 320 exemplaires.

Avec notre réseau ainsi obtenu, nous sommes parvenu à un taux de reconnaissance de 78.75 %.

Mots Clefs

ESC-10, réseau de neurones convolutif, reconnaissance audio

Abstract

In an inevitable connected world, the automatic recognition of environmental sounds can be really interesting, for example in the medical field to help the hearing impaired but also in the field of robotics where we try more and more to reproducing human faculties. We would like to automatically recognize mutually exclusive environmental sounds and identify a particular noise in an audio sequence. We used here the ESC-10 corpus specially designed to classify 10 different environmental sounds. Unfortunately, the networks of neurons processing the sound do not exist at the moment we choose to convert the audio sequences into spectrograms in order to have a spectral representation of the signal that we will analyze as an image. We have used different convolutions, followed alternately by a Tanh activation function or a ReLU. We have linked it with one dense Tanh layer with dropout. To distinguish 10 sounds in exclusive ways: we have a final 10-neuron layer with the softmax activation function. We trained our network on a corpus of 320 copies. With our network thus obtained, we have achieved a recognition rate of 78.75%.

Keywords

ESC-10, convolutional neural network, audio recognition.

1 Introduction

L'avancée technologique étant de plus en plus présente mais aussi les ordinateurs étant de plus en plus puissants. Il peut être intéressant de savoir identifier dans une séquence audio quel est le bruit environnant. Par exemple si l'on cherche à automatiser des générations de sous-titres malentendants. Des recherches ont déjà été élaborées pour résoudre ce problème, c'est pourquoi il existe déjà une base de données existante et des résultats expérimentaux avec lesquels nous pourrons comparer nos résultats. Dans un premier temps nous allons présenter le dataset utilisé, suivi de notre modèle mis en place (le prétraitement des données à l'aide de réseaux convolutifs, puis les couches denses et de sorties). Enfin nous présenterons nos résultats obtenus.

1.1 Présentation du dataset

Nous avons utilisé le jeu de données ESC-10 composé d'un ensemble de 400 enregistrements étiquetées en 10 classes (40 enregistrements de chaque classes) réalisé par

K. J. Piczak [1][2]. Il classifie ainsi respectivement le bruit d'une tronçonneuse, d'un tic-tac d'une horloge, d'un craquement de feu, de pleurs de bébé, d'un chien, d'un hélicoptère, de la pluie, d'un coq, d'un bruit des vagues et d'un éternuement. Il s'agit d'un sous-ensemble d'un plus grand jeu de données : ESC-50 [1].
Ce jeu de données permet d'analyser des bruits très variés puisqu'en effet il regroupe trois grands types de sons différents : certains avec des patterns temporels caractéristiques, certains avec de fortes harmoniques ainsi que certains sons plus ou moins structurés [1].

2 Présentation du modèle implémenté

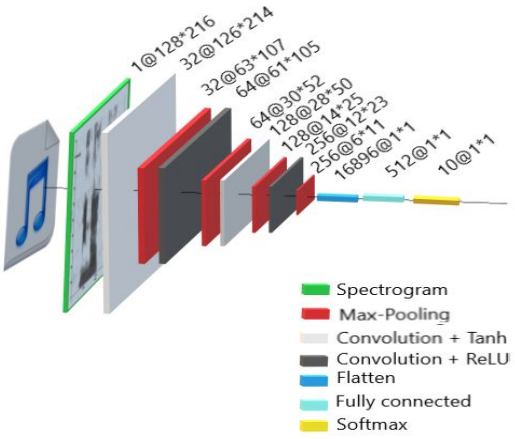


figure 1 : Représentation du modèle utilisé.

2.1 Données en entrées

Les fichiers audios sont convertis en spectrogrammes, images 2D représentant l'énergie du spectre de fréquences d'un son qui varie dans le temps [3]. Ils sont très fréquemment utilisés pour repérer les zones caractéristiques dans les bandes sonores [4]. Les spectrogrammes donnent une représentation 2D (figure 2) que nous pouvons par la suite analyser par un réseau convolutif.
Nous avons prétraité les données en encodant les labels de sortie sous forme de vecteur "one-hot" et normalisé nos données en utilisant la standardisation.

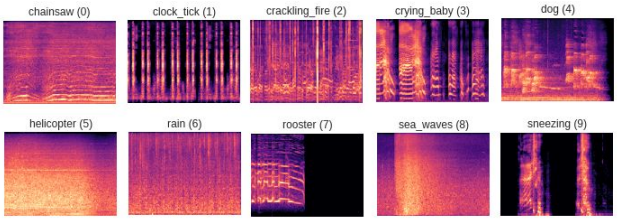


figure 2 : Spectrogrammes obtenus des premiers enregistrements de chaque classe.

2.2 Couche convolutif

Le réseau convolutif employé est constitué de 4 couches de convolution avec des filtres de taille fixée à 3x3, chacune séparée par un Max-Pooling (figure 1). Les entrées étant à la base des signaux sinusoïdaux et la fréquence ayant une allure sinusoïdale, nous avons trouvé intéressant d'alterner nos fonctions d'activations entre Tanh et ReLU. La fonction ReLU (Unité Linéaire Rectifiée) étant souvent choisie pour des couches d'activations en raison de sa convergence rapide [4]. La fonction Tanh se rapprochant plus des fonctions sinusoïdales liées à la représentation fréquentielle.

2.3 Couche dense (fully connected)

Après aplatissement de la dernière couche convolutive, nous avons introduit une couche dense de taille 512 avec une fonction d'activation Tanh et un dropout de 80%, ce dernier permettant de limiter le surapprentissage [5].

2.4 Couche de sortie

Le réseau devant reconnaître 10 classes exclusives, nous avons terminé notre réseau par une couche dense de 10 neurones avec une softmax permettant d'avoir une certaine probabilité que le fichier audio donné en entrée appartienne à une classe spécifique (figure 3). Afin d'analyser notre taux de précision, nous avons choisi d'utiliser une fonction de coût qui est la categorical cross-entropy (figure 4).

$$softmax(z_i) = \frac{exp(z_i)}{\sum_j exp(z_j)}$$

figure 3 : Fonction softmax [6]

$$\mathcal{L}(y - \hat{y}) = -\sum_{i=1}^{n} y_i \log \hat{y}_i$$

figure 4: Fonction de perte "categorical cross-entropy" [7]

Tout le long de notre modèle, la fonction d'optimisation utilisée a été l'optimizer adagrad (avec un pas d'apprentissage de 3.10^{-3}) (figure 5).

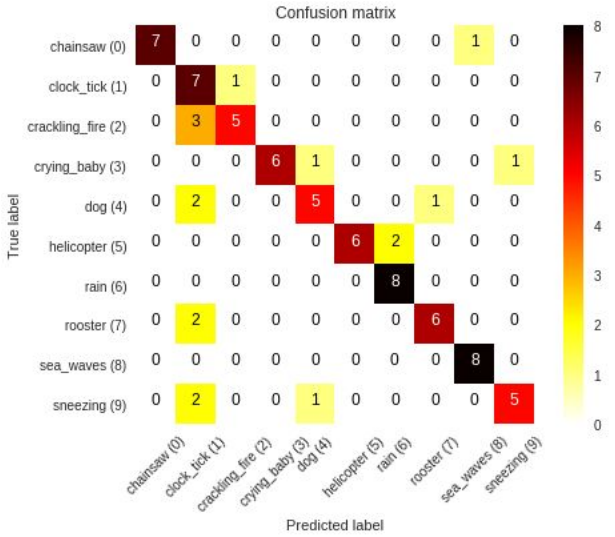
$$\theta_{t+1,i} = \theta_{t,i} - \frac{\eta}{\sqrt{G_{t,ii} + \epsilon}} \cdot g_{t,i}$$

figure 5: Fonction de l'optimiseur "adagrad" [8]

3 Résultats

D'après [1], le jeu de données ESC-10 a été testé préalablement avec des techniques d'apprentissage automatique plus communes tel que l'algorithme des k-plus proches voisins avec un taux de réussite de 66.7%, les séparateurs à vaste marge à 67,5% ou encore les forêts d'arbres décisionnels qui ont réussi à atteindre 72.7%. Nous restons tout de même assez éloignés des meilleurs scores proposés avec le réseau GoogleNet à 91.3% de précision dans [3][9].

Dans notre cas, nous réussissons à obtenir en moyenne de 75% de précision avec un pic à 78.75% selon les neurones retirés via le dropout. Le tout étant sur 100 epochs avec un mini batch de 32. Voici une des matrices de confusions que nous avons obtenu.




```
chainsaw: 0.8750          helicopter: 0.7500
clock_tick: 0.8750        rain: 1.0000
crackling_fire: 0.6250    rooster: 0.8750
crying_baby: 0.7500       sea_waves: 1.0000
dog: 0.6250               sneezing: 0.7500
```

figure 6 : Matrice de confusion + résultats par classe

Nous pouvons constater qu'il a pu totalement reconnaître un son continu de fréquence uniforme tel que la pluie. On remarque aussi qu'il différencie moins bien le crépitement du feu avec le tic tac d'horloges. En écoutant le corpus, les deux sons étant assez proches, cette erreur semble compréhensible.

On peut observer une courbe de la fonction perte (loss) qui diminue bien à chaque epochs jusqu'à atteindre un seuil en dessous de 0.15. (figure 7)

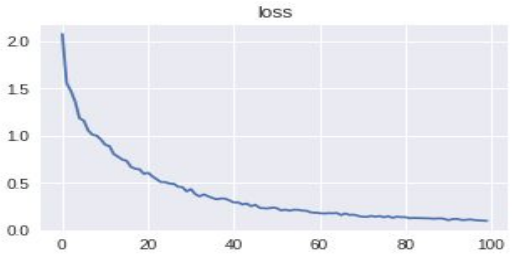


figure 7 : courbe de la fonction de perte

Nous pouvons également observer la courbe accuracy qui augmente bien à chaque epochs. L'irrégularité de la courbe peut s'expliquer par notre ajout d'un dropout, lequel a pour effet d'augmenter la variance. (figure 8)

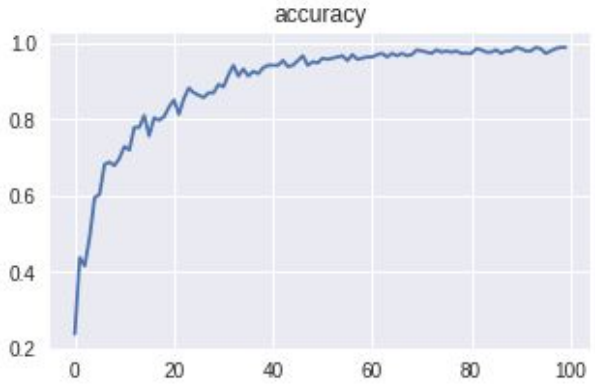


figure 8 : courbe du taux de précision

Conclusion

Nous avons pu implémenter un modèle de réseau de neurones pour classifier différents types de sons environnants et obtenir un taux d'accuracy jusqu'à 78.75%.

Pour comparer nous avons aussi effectué un MLP de 768 neurones qui arrive difficilement à dépasser 37.5% re reconnaissance.

Nous avons aussi pensé qu'il aurait pu être intéressant de fusionner notre réseau avec un RNN (Réseau de Neurones Récurrents, LSTM ou GRU) qui permettrait notamment de conserver une certaine notion du temps. En faisant nos recherches nous avons constaté qu'une étude a déjà été établi dessus sur ce dataset mais que les résultats n'étaient malheureusement pas probants [3].

Durant nos recherches nous avons pu remarquer que les meilleurs réseaux implémentés pour ce type de jeu de données reposent beaucoup sur un meilleur prétraitement des données (variance de la taille de la trame).

Remerciements

Nous remercions Thomas Pelligrini ainsi que Benjamin Chamand qui nous ont introduit ce problème avec sa structure pour nous introduire le concept des réseaux de neurones convolutifs sur des fichiers sonores.

Bibliographie

[1] K. J. Piczak, “ESC: Dataset for environmental sound classification,” in Proceedings of the ACM International Conference on Multimedia. ACM, 2015, in press.

[2] Karol J. Piczak. ESC: Dataset for Environmental Sound Classification. https://doi.org/10.7910/DVN/YDEPU , Harvard Dataverse, 2015.

[3] Boddapati, Venkatesh, “Classifying Environmental Sound with Image Networks,” in Faculty of Computing. Blekinge Institute of Technology, 2017.

[4] E. Vincent, “Séparation de signaux audio : principes statistiques de l’analyse en composantes indépendantes et applications au signal monophonique,” in Rapport de stage de DEA ATIAM. IRCAM, 2001.

[5] N. Srivastava et al., “Dropout: A Simple Way to Prevent Neural Networks from Overfitting,” in Department of Computer Science. University of Toronto, 2014.

[6] N. E. West, T. J. O’Shea, “Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition,” U.S. Naval Research Laboratory, 2017. https://arxiv.org/pdf/1703.09197.pdf

[7] K. J. Piczak, “Deep Architectures for Modulation Recognition,” in Proceedings of the ACM International Conference on Multimedia. ACM, 2015, in press.

[8] A. S. Waliaimage, “Types of Optimization Algorithms used in Neural Networks and Ways to Optimize Gradient Descent” , Towards Data Science. 2017. https://towardsdatascience.com/types-of-optimization-algorithms-used-in-neural-networks-and-ways-to-optimize-gradient-95ae5d39529f

[9] V. Boddapati, A. Petef et al., “Classifying environmental sounds using image recognition networks,” in International Conference on Knowledge Based and Intelligent Information and Engineering Systems. KES2017, 2017.