Clasificarea păsărilor utilizând Ensemble Classifiers

1. Introducere

LifeCLEF 2014 Birt Task (BirdCLEF) conține 14027 înregistrări audio cu 501 specii de păsări, mult mai mult în comparație cu competiția anuala MLSP sau NIPS4B. Adunat, baza de date MLSP și a NIPS4B ocupă aproximativ 1GB în timp ce BirdCLEF ocupa peste 25GB.

Luând în considerare diferența bazei de date și numărul de specii ce trebuie identificate, algoritmii folosiți în MLSP și NIPS4B ar fi ineficienți.

Toate experimentele BirdCLEF sunt realizate pe desktop-uri standard cu procesoare 4-core si 8GB RAM așadar există constrângeri severe legate de metodele algoritmilor de tip machine learning și am ales să ne îndreptăm atenția asupra algoritmilor genetici simpli cu funcții simple și care găsesc o soluție optimă.

1. Front-end

Folosim atât date audio cât și meta date deoarece ofera informații complementare clasificatorului. Memoria utilizată în computaționare depinde de mărimea caracteristicilor.

* 1. MFCC

Atributele MFCC oferite de organizator conțin un singur coeficient de energie și 15 coeficienți cepstral. Fiecare fisier audio se segementează renunțând la toate frame-urile cu coeficientul sub medie. Cadrele rămase sunt sortate după numărul de cadre ce ajunge la final.

* 1. Caracteristicile spectral

Fișierele audio sunt convertite în spectograme, reprezentând sunetul prin suprapunerea ferestrelor de 200ms la interval de 100ms. Dimensiunea frecvenței este calculată prin medie, acesta este elimintă la fel ca în MFCC. În final, algoritmul este aplicat în n dimensiuni spectrale pentru a îi reduce intervalul dinamic.

* 1. Caracteristici meta

Pe langă datele audio descrise mai sus, mai folosim și: latitudinea, longitudinea, elevarea, anul, luna, luna+zi, timpul și autorul.

Câmpul LUNA + ZI combină luna și ziua într-un singur câmp de 4 biți deoarece compararea aceleași zile în luni diferite este ilogică.

În comparație cu MLSP, BirdCLEF nu obligă completarea datelor într-un anumit format. Informațiile necunoscute sau lipsă trebuiesc parsate manual în numere.

1. Back-end

Deși etichetele oficiale returnează o singură specie de pasăre dată de ClassID, câmpul BackgroundSpecies listează deobicei specii adiționale, oricum, la fel ca meta datele, acest câmp este neconcludent deoarece nu există indicații cum că există specii adiționale.

Altă problemă derivată din cea precedentă este prezența altor specii ce nu se găsește în cele 517 specii clasificate, din care 16 am ales să le ignorăm deoarece nu există destule informații încât să poată fi identificate.

* 1. Python

Clasificatorii folosiți in Python sunt construiți cu ajutorul suitei scikit-learn. Pentru a construi un ansamblu de clasificatori, cerem acestora să afișeze probabilitatea estimată pentru fiecare clasă pentru a calcula media individuală. Dintre clasificatorii disponibili am ales următoarea listă:

**ExtraTreesClassifier, RandomForestClassifier,KNeighborsClassifier, LogisticRegression, SGDClassifier, AdaBoostClassifier, GradientBoostingClassifier, SVC, GaussianNB, BernoulliNB, LDA**

Din lista de mai sus, doar ExtraTreesClassifier, RandomForestClassifier și K NeightborsClassifier sunt capabile să controleze clasificarea multi-eticheta. OneVsRestClassifier este folosit pentru a transforma clasificatorii binari rămași în clasificatori multi-label multi-class.

3.4 Matlab

Trei clasificatori sunt folositi in subsistemul Matlab : Random Forests folosing functia TreeBagger si LDA folosind ClassificationDiscriminant.fit , ambele antrenate intr-o configuratie unu vs restul. Ambele clasificatoare afiseaza estimari prin functia predict si outputul final este media dintre clasificatoarele individuale. Functia TreeBagger a fost modificata in asa fel incat fiecare arbore din padure a fost antrenat cu un set echilibrat de date negative , si asta a inbunatatit semnificativ clasificarea fara a afecta performanta. In mod particular fiecare arbore a fost antrenat cu toate datele din clasa pozitiva si cu o submultime din datele negative continand de 20 de ori mai multe date decat cele pozitive. Numarul variabilelor de proba la fiecare nod au fost setate la 3 in timp ce numarul de arbori a fost setat la 200 pentru rularea finala de evaluare.

1. Setup-ul Experimentului

Pentru a evalua performanta si combinatiile clasificatorului, sau dat seturi de date continand 10 iteme cunoscute si 10 iteme necunoscute.

4.1 Subsistemul Python

Din lista de caracteristici front end si clasificatori selectati din sklearn , combinatii dintre clasficari si caracteristici sunt evaluate. Acest pas a consumat foarte mult din timpul disponibil datorita optimizarii setarilor caracteristicilor si clasificatorilor si a combinatiilor dintre cele doua.

4.2 Subsistem Matlab

O singura configuratie de caracteristici este folosita in Matlab. Folosing acceasi terminologie ca si pana acum , putem numi caracteristicile ca Meta + Spec40, cu n=40 dimensiuni folosite pentru caracteristicile spectrale concatenate cu cele 8 dimensiuni ale caracteristicilor meta. Deosebirea fata de subsistemul Python este ca , caracteristicile spectrale locale sunt generate luand maximul din intervalele selectate local. Fiecare interval este jumatate de secunda, si un pas este echivalent cu incrementarea indexului intervalului. Energia fiecarui interval este calculata ca si media logaritmica a energiei in interval si apoi este folosit ca si metric pentru selectarea intervalelor in clasificare. Doar intervalele care reprezinta maximul energic au fost pastrate. Ideea este ca intervalele continand maximul din energie vor corespunde instantele locale ale cantecelor pasarilor din clip.

1. Rezultate

|  |  |  |  |
| --- | --- | --- | --- |
|  | Python | Matlab | Combined |
| Cu specii de fundal | 0.284 | 0.166 | 0.289 |
| Fara specii de fundal | 0.267 | 0.159 | 0.272 |
|  |  |  |  |

Sistemul create a fost antrenat pentru a recunoaste mai multe specii dintr-o singur clip de aceea rezultatele sunt mult mai bune ca si scor pe fisierele cu specii in fundal, decat pe cele fara.

1. Concluzii

Dificultatile in challenge sunt legate de numarul mare de date, de numarul mare de specii tinta ce trebuie clasificate, meta date gresite sau incomplete. De asemenea timpul limitat si etichetele multi clasa gresite au facut challenge-ul dificil.