

# Sastanak Lumen no.1 - 12.2

Za konačni model bi mogli uzeti ansamble

- znači kombiniramo više modela
  - glavni model na osnovu spektograma -> neki vision model (npr. pretrenirani vision transformeri, postoje već neki koji rade sa spektogramima)
  - a za ostale modele bi mogli uzeti modele koji rade sa "čistim" zvukom
    - naravno, prije toga neki preprocessing da izdvojimo neke zanimljive značajke i slično
    - istražiti što postoji

Pogledati postojeća rješenja u analizi zvuka, vidjeti što možemo primijeniti

- vidjeti što ljudi rade s klasifikacijom zvuka

Dinamičko generiranje podataka obavezno

- da izbjegnemo kombinatornu eksploziju a i za lakše treniranje

Riješiti problem ulazne rezolucije spektograma

- što je dulji sample, spektogram će biti lošije kvalitete
  - jer ćemo na istoj rezoluciji prikazivati više podataka
- također, treba skužiti kako generirati i spremiti "čisti" spektogram (bez "legende" sa strane) , i vidjeti može li mu se podesiti rezolucija

Gdje trenirati model/gdje pokretati npr. jupyter notebooke

- za jupyter notebooke colab je ok rješenje
- za treniranje modela
  - colab pa hrpa checkpointova
  - eventualno na zeso server

Kromogram izgleda fora, njega bi također možda mogli klasificirati vision transformerom

- pošto su to doslovno note
- ali spektogram je i dalje glavni

Što se tiče featura "žanr", nije baš lako za koristiti jer nemamo taj podatak u val/test setu

## Potrebne odluke

- koji tip spektograma koristimo
- koristimo li filtere (i koje ako da) prije nego što generiramo spektrogram
- također, je li potreban još neki dodatan preprocessing
- kako napraviti dinamičko sampleiranje

Bitno -> prilikom istraživanja i implementacije voditi bilješke, kako bi kasnije mogli lakše napisati dokumentaciju

---

do idućeg sastanka -> pogledati postojeće radove

## Roadmap

- 1) pogledati postojeće radove, što ljudi rade u području audio klasifikacije
  - vidjeti koje bi još feature možemo uzeti
- 2) napraviti skriptu za dinamičko učitavanje podataka
- 3) kreirati osnovni model samo na osnovu spektograma
  - za početak to može biti samo proof of work, ala VGG/ResNet na manjem broju primjera
    - treniranje tog modela može ići preko colaba
  - a kasnije kad vidimo da sve radi uzmemo nešto složenije (npr. vision transformers i slično)
    - za ovo smisliti kako trenirati
- 4) istrenirati "napredniji" model nad spektrogramima
- 5) istražiti i implementirati dodatne metode
  - npr. kromogrami, čisti zvuk, dodatne značajke i slično
  - tu se možemo podijeliti, kad budemo imali ideju što točno hoćemo uzeti
- 6) kreirati ansambl modela, usporediti rezultate s pojedinačnim modelima
- 7) napraviti REST API
- 8) napraviti neki jednostavan frontend
- 9) napisati dokumentaciju
  - primjeri dokumentacije
    - <https://github.com/karlic-luka/LUMEN-Data-Science-2020/blob/main/Dokumentacija.pdf>
    - <https://github.com/PatrikDurdevic/LUMEN-Data-Science-2020-Predictive-Maintenance/tree/master/Solution%20Documentation>
    - <https://github.com/josipdomazet/lumen2021>

- <https://github.com/ciglencecki/geoguessr-ai/blob/main/docs/documentation.md> --> prošla godina
- 10) ???
- 11) profit

opcionalan korak -> deployati app na javni server/exposati na internet

- mislim da imamo dosta toga free na microsoft azuereu u sklopu fer acca
- nije spomenuto nigdje, al potencijalno je plus u odnosu na ostale timove