UNIVERSITATEA TEHNICĂ
DIN CLUJ-NAPOCA

Facultatea de Automatică și Calculatoare

Clasificarea sunetelor pulmonare și detecția anomaliilor folosind date audio obținute cu stetoscopul

Proiect de semestru

Disciplina: **Sisteme bazate pe cunoaștere**

Student: Sabău Alexandra-Denisa

An universitar: 2025–2026

Cuprins

1. Introducere

1.1 Context general

Sunetele respiratorii reprezintă o sursă importantă de informație în diagnosticul bolilor pulmonare. În practica medicală, acestea sunt analizate în mod tradițional prin auscultație, o metodă care depinde în mare măsură de experiența și subiectivitatea medicului. Diferențele subtile dintre sunetele normale și cele patologice pot fi dificil de identificat, mai ales în condiții clinice aglomerate.

Dezvoltarea tehnologiilor de procesare a semnalelor și a metodelor de învățare automată a permis apariția unor sisteme capabile să analizeze automat semnale audio medicale. În special, utilizarea rețelelor neuronale convoluționale (CNN), împreună cu reprezentări timp–frecvență precum spectrogramele, a demonstrat rezultate promițătoare în recunoașterea tiparelor complexe din date audio.

În contextul sistemelor bazate pe cunoaștere, aceste modele pot fi considerate mecanisme de extragere automată a cunoașterii din date, învățând relații implicite între caracteristicile semnalului audio și starea de sănătate a pacientului. Astfel de sisteme pot oferi suport decizional în diagnostic și pot contribui la detectarea timpurie a anomaliilor respiratorii.

1.2 Obiectivele proiectului

Obiectivul principal al acestui proiect este realizarea unui sistem automat pentru analiza sunetelor respiratorii, capabil să diferențieze între starea normală și diverse patologii pulmonare, utilizând metode moderne de deep learning.

Obiectivele specifice ale proiectului sunt:

- analiza și înțelegerea unui set de date audio medicale;
- pre-procesarea semnalelor audio și transformarea acestora în reprezentări adecvate pentru învățare automată;
- extragerea caracteristicilor relevante folosind spectrograme log-Mel;
- construirea și antrenarea unui model CNN pentru clasificare multi-clasă;

- evaluarea performanței modelului folosind metrici standard;
- reformularea problemei ca detecție de anomalii (Normal vs Abnormal) și analiza performanței obținute.

1.3 Specificații generale

Sistemul a fost implementat folosind limbajul de programare Python și biblioteci consacrate pentru procesarea semnalelor audio și dezvoltarea modelelor de deep learning. Datele de intrare sunt reprezentate de fișiere audio în format `.wav`, iar ieșirea sistemului constă în diagnosticul prezis pentru fiecare înregistrare.

Principalele limitări ale sistemului sunt legate de dezechilibrul setului de date și de numărul redus de exemple pentru anumite clase patologice, aspecte care influențează performanța clasificării multi-clasă. Aceste limitări sunt analizate și discutate în capitolele dedicate evaluării performanței.

2. Cunoașterea și analiza setului de date

2.1 Descriere generală a setului de date

Setul de date utilizat în cadrul acestui proiect este format din înregistrări audio ale sunetelor respiratorii, provenite de la pacienți cu diverse afecțiuni pulmonare, precum și de la pacienți sănătoși. Fiecare fișier audio reprezintă o înregistrare a respirației unui pacient și este asociat unui diagnostic medical.

Datele provin din două surse principale:

- un set de date original, care conține fișiere audio în format `.wav` și un fișier CSV ce asociază fiecărui pacient un diagnostic;
- un set de date suplimentar, organizat pe foldere, fiecare folder corespunzând unei clase patologice sau stării de sănătate.

Diagnosticele identificate în cadrul setului de date includ, printre altele, COPD, Pneumonia, Healthy, URTI, Bronchiectasis și Bronchiolitis. Această varietate de clase permite evaluarea capacității sistemului de a diferenția între starea normală și mai multe tipuri de patologii respiratorii.

```
Distribuție pe clase înainte de filtrare:

diagnosis
COPD              887
Pneumonia         111
Healthy           103
URTI               21
Bronchiectasis     16
Bronchiolitis      13
LRTI                2
Asthma              1
Name: count, dtype: int64

Clase păstrate (>=3 exemple):
['COPD', 'Pneumonia', 'Healthy', 'URTI', 'Bronchiectasis', 'Bronchiolitis']

Distribuție pe clase DUPĂ filtrare:

        diagnosis  count
0            COPD    887
1       Pneumonia    111
2         Healthy    103
3            URTI     21
4  Bronchiectasis     16
5   Bronchiolitis     13

Număr total de înregistrări după filtrare: 1151
```

Figura 2.1: Distribuția claselor înainte și după filtrarea claselor rare

Analiza distribuției claselor evidențiază un dezechilibru semnificativ al setului de date. Unele clase, precum *COPD*, sunt puternic reprezentate, în timp ce altele conțin un număr foarte redus de exemple. Această situație este frecvent întâlnită în seturile de date medicale și reprezintă o provocare majoră pentru antrenarea modelelor de clasificare multi-clasă.

Pentru a asigura stabilitatea procesului de antrenare și posibilitatea împărțirii corecte a datelor în seturi de antrenare, validare și test, clasele cu mai puțin de trei exemple au fost eliminate. În urma acestei filtrări, setul de date final conține 1151 de înregistrări audio și șase clase distincte.

Dezechilibrul rămas între clase trebuie luat în considerare în interpretarea rezultatelor obținute, influențând în special performanța clasificării multi-clasă pentru clasele slab reprezentate.

3. Procesarea setului de date

3.1 Pre-procesarea semnalelor audio

Semnalele audio brute reprezintă variații ale amplitudinii în timp și pot avea durate și niveluri de energie diferite de la o înregistrare la alta. Pentru a putea fi utilizate într-un sistem de învățare automată, aceste semnale necesită o etapă de procesare preliminară. Fiecare fișier audio este încărcat și normalizat, astfel încât diferențele de amplitudine dintre înregistrări să fie reduse și informația relevantă să fie comparabilă între exemple.

Deoarece rețelele neuronale convoluționale sunt concepute pentru a procesa date de tip imagine, semnalul audio este transformat într-o reprezentare timp–frecvență sub forma unei spectrograme. Spectrograma Mel descrie modul în care energia semnalului este distribuită în timp și pe benzi de frecvență, fiecare pixel al imaginii corespunzând unei valori de energie pentru o anumită frecvență, la un moment de timp.

Pentru a evidenția mai bine diferențele subtile dintre sunetele normale și cele patologice, spectrograma Mel este convertită în scară logaritmică, rezultând spectrograma Log-Mel. Această transformare comprimă intervalul mare de valori energetice și scoate în evidență variațiile fine ale semnalului, facilitând învățarea tiparelor relevante de către modelul CNN.

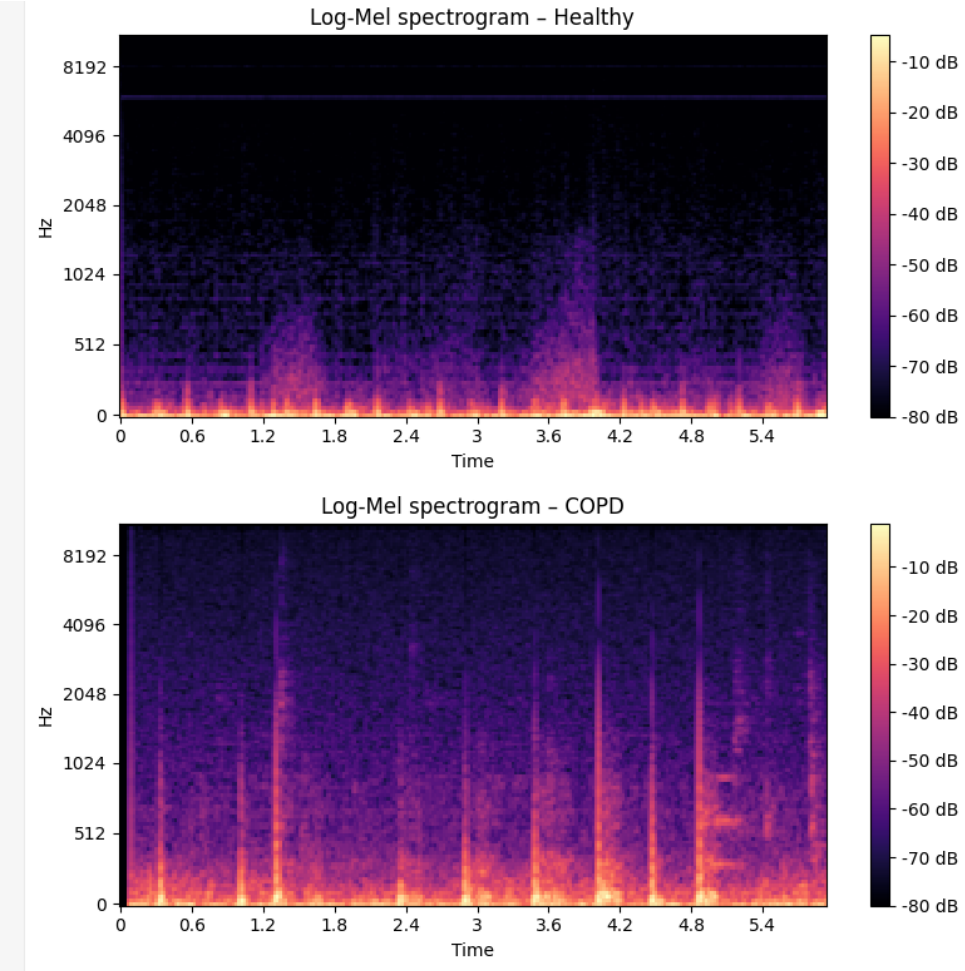

Figura 3.1: Exemple de spectrograme Log-Mel pentru clasele Healthy și COPD

Figura 3.1 ilustrează diferențele dintre un sunet respirator normal și unul patologic. Spectrograma Healthy prezintă o distribuție energetică relativ uniformă, în timp ce spectrograma COPD evidențiază variații neregulate și componente frecvențiale distincte.

3.2 Standardizarea dimensiunii intrării

Pentru a permite utilizarea spectrogramelor ca intrare într-o rețea neuronală convoluțională, toate reprezentările Log-Mel sunt aduse la aceeași dimensiune. Spectrogramele sunt completate cu zerouri sau trunchiate astfel încât dimensiunea finală să fie de 128 benzi Mel și 256 pași temporali. Această standardizare permite aplicarea metodelor de învățare profundă utilizate în procesarea imaginilor asupra semnalelor audio medicale.

4. Modelarea sistemului

4.1 Reprezentarea datelor

După etapa de procesare, fiecare înregistrare audio este transformată într-o reprezentare numerică sub forma unui tensor 4D, corespunzător unei spectrograme Log-Mel de dimensiune fixă (128 × 256, cu un singur canal). Această formă este necesară pentru a putea fi utilizată ca intrare într-o rețea neuronală convoluțională.

Diagnosticele asociate fiecărei înregistrări sunt convertite în valori numerice cu ajutorul unui codificator de etichete, iar pentru procesul de antrenare al modelului sunt reprezentate în format one-hot. Setul de date este apoi împărțit în subseturi de antrenare, validare și testare, folosind o strategie stratificată, astfel încât proporția claselor să fie păstrată în fiecare subset.

```
Clasele modelului: ['Bronchiectasis' 'Bronchiolitis' 'COPD' 'Healthy' 'Pneumonia' 'URTI']
Număr de clase: 6
X_all shape: (1151, 128, 256, 1)
y_all shape: (1151,)
Train: (805, 128, 256, 1) Val: (173, 128, 256, 1) Test: (173, 128, 256, 1)
```

Figura 4.1: Structura seturilor de date utilizate în procesul de modelare

4.2 Arhitectura modelului CNN

Modelul utilizat este o rețea neuronală convoluțională care tratează spectrogramele Log-Mel ca imagini alb-negru. Arhitectura este formată din:

- trei blocuri convoluționale, fiecare conținând un strat Conv2D, un strat Batch-Normalization și un strat MaxPooling;
- un strat Flatten pentru transformarea hărților de caracteristici într-un vector;
- două straturi Dense separate de straturi Dropout pentru prevenirea supraînvățării;
- un strat de ieșire cu funcția de activare softmax, corespunzător clasificării multi-clasă.

Layer (type)	Output Shape	Param #
input_layer (InputLayer)	(None, 128, 256, 1)	0
conv2d (Conv2D)	(None, 128, 256, 16)	160
batch_normalization (BatchNormalization)	(None, 128, 256, 16)	64
max_pooling2d (MaxPooling2D)	(None, 64, 128, 16)	0
conv2d_1 (Conv2D)	(None, 64, 128, 32)	4,640
batch_normalization_1 (BatchNormalization)	(None, 64, 128, 32)	128
max_pooling2d_1 (MaxPooling2D)	(None, 32, 64, 32)	0
conv2d_2 (Conv2D)	(None, 32, 64, 64)	18,496
batch_normalization_2 (BatchNormalization)	(None, 32, 64, 64)	256
max_pooling2d_2 (MaxPooling2D)	(None, 16, 32, 64)	0
flatten (Flatten)	(None, 32768)	0
dropout (Dropout)	(None, 32768)	0
dense (Dense)	(None, 128)	4,194,432
dropout_1 (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 6)	774

Total params: 4,218,950 (16.09 MB)

Trainable params: 4,218,726 (16.09 MB)

Non-trainable params: 224 (896.00 B)

Figura 4.2: Sumarul arhitecturii rețelei neuronale convoluționale

Figura prezintă arhitectura rețelei neuronale convoluționale utilizate în acest proiect, precum și dimensiunea ieșirii și numărul de parametri pentru fiecare strat. Modelul primește ca intrare spectrograme Log-Mel tratate ca imagini alb-negru, cu dimensiunea 128×256 și un singur canal.

Primul strat convoluțional aplică 16 filtre de dimensiune 3×3 asupra spectrogramelor de intrare. Aceste filtre au rolul de a identifica tipare locale simple, precum variații de energie, margini sau tranziții bruște în timp și frecvență. Dimensiunea spațială a datelor este păstrată datorită utilizării padding-ului de tip „same”. Stratul de normalizare BatchNormalization stabilizează distribuția activărilor și accelerează procesul de învățare, iar stratul MaxPooling reduce rezoluția spațială, păstrând în același timp cele

mai relevante informații.

Al doilea bloc convoluțional extinde numărul de filtre la 32, permițând modelului să combine tiparele simple detectate anterior în structuri mai complexe. Prin aplicarea succesivă a convoluției, normalizării și pooling-ului, modelul începe să surprindă relații mai subtile între componentele temporale și frecvențiale ale spectrogramelor.

Al treilea bloc convoluțional crește numărul de filtre la 64 și continuă procesul de abstractizare a informației. În această etapă, rețeaua învață reprezentări de nivel înalt, care pot fi asociate cu tipare specifice diferitelor patologii respiratorii. Reducerea progresivă a dimensiunii spațiale și creșterea adâncimii permit captarea informației relevante într-o formă compactă și expresivă.

După etapa de extracție a caracteristicilor, stratul Flatten transformă hărțile de caracteristici într-un vector unidimensional, pregătind datele pentru procesul de decizie. Straturile Dense care urmează realizează clasificarea propriu-zisă, combinând caracteristicile extrase pentru a produce o predicție finală. Utilizarea straturilor Dropout în această etapă reduce riscul de supraînvățare, forțând modelul să nu se bazeze excesiv pe un număr redus de neuroni.

Stratul de ieșire utilizează funcția de activare softmax și are un număr de neuroni egal cu numărul claselor. Acesta produce o distribuție de probabilitate peste clasele posibile, permițând alegerea diagnosticului cu probabilitatea cea mai mare.

În ansamblu, arhitectura este concepută pentru a extrage treptat informația relevantă din spectrograme, pornind de la caracteristici locale simple și ajungând la reprezentări complexe, adecvate pentru clasificarea sunetelor respiratorii.

5. Antrenarea modelului

5.1 Procedura de antrenare

Modelul este antrenat utilizând optimizerul Adam și funcția de pierdere categorical crossentropy. Procesul de antrenare se desfășoară pe parcursul a 20 de epoci, cu un batch size de 16. Performanța este monitorizată folosind setul de validare.

```
Epoch 1/20
51/51 ━━━━━━━━━━━━━━━━━━━━ 35s 550ms/step - accuracy: 0.6323 - loss: 4.1622 - val_accuracy: 0.7746 - val_loss: 20.0889
Epoch 2/20
51/51 ━━━━━━━━━━━━━━━━━━━━ 37s 480ms/step - accuracy: 0.7491 - loss: 0.7870 - val_accuracy: 0.6012 - val_loss: 3.2375
Epoch 3/20
51/51 ━━━━━━━━━━━━━━━━━━━━ 40s 452ms/step - accuracy: 0.7466 - loss: 0.7248 - val_accuracy: 0.6879 - val_loss: 3.1021
Epoch 4/20
51/51 ━━━━━━━━━━━━━━━━━━━━ 26s 514ms/step - accuracy: 0.7627 - loss: 0.6840 - val_accuracy: 0.7803 - val_loss: 0.6260
Epoch 5/20
51/51 ━━━━━━━━━━━━━━━━━━━━ 38s 458ms/step - accuracy: 0.7602 - loss: 0.6189 - val_accuracy: 0.7746 - val_loss: 0.5953
Epoch 6/20
51/51 ━━━━━━━━━━━━━━━━━━━━ 28s 555ms/step - accuracy: 0.7801 - loss: 0.5667 - val_accuracy: 0.7746 - val_loss: 2.7997
Epoch 7/20
51/51 ━━━━━━━━━━━━━━━━━━━━ 1176s 24s/step - accuracy: 0.7814 - loss: 0.5576 - val_accuracy: 0.7746 - val_loss: 10.9163
Epoch 8/20
51/51 ━━━━━━━━━━━━━━━━━━━━ 27s 536ms/step - accuracy: 0.7888 - loss: 0.4989 - val_accuracy: 0.7630 - val_loss: 0.7479
Epoch 9/20
51/51 ━━━━━━━━━━━━━━━━━━━━ 30s 588ms/step - accuracy: 0.7901 - loss: 0.5693 - val_accuracy: 0.7746 - val_loss: 0.8276
Epoch 10/20
51/51 ━━━━━━━━━━━━━━━━━━━━ 28s 536ms/step - accuracy: 0.8012 - loss: 0.4616 - val_accuracy: 0.7746 - val_loss: 0.7973
Epoch 11/20
51/51 ━━━━━━━━━━━━━━━━━━━━ 26s 515ms/step - accuracy: 0.8087 - loss: 0.4643 - val_accuracy: 0.8439 - val_loss: 0.6191
Epoch 12/20
51/51 ━━━━━━━━━━━━━━━━━━━━ 26s 517ms/step - accuracy: 0.8335 - loss: 0.4146 - val_accuracy: 0.7746 - val_loss: 11.7355
Epoch 13/20
51/51 ━━━━━━━━━━━━━━━━━━━━ 26s 510ms/step - accuracy: 0.8447 - loss: 0.3898 - val_accuracy: 0.7861 - val_loss: 0.5872
Epoch 14/20
51/51 ━━━━━━━━━━━━━━━━━━━━ 28s 546ms/step - accuracy: 0.8335 - loss: 0.3833 - val_accuracy: 0.8844 - val_loss: 0.3818
Epoch 15/20
51/51 ━━━━━━━━━━━━━━━━━━━━ 131s 3s/step - accuracy: 0.8460 - loss: 0.3967 - val_accuracy: 0.7746 - val_loss: 3.0325
Epoch 16/20
51/51 ━━━━━━━━━━━━━━━━━━━━ 28s 551ms/step - accuracy: 0.8050 - loss: 0.5514 - val_accuracy: 0.8035 - val_loss: 1.0691
Epoch 17/20
51/51 ━━━━━━━━━━━━━━━━━━━━ 26s 517ms/step - accuracy: 0.8137 - loss: 0.4291 - val_accuracy: 0.8497 - val_loss: 0.4435
Epoch 18/20
51/51 ━━━━━━━━━━━━━━━━━━━━ 27s 532ms/step - accuracy: 0.8273 - loss: 0.4188 - val_accuracy: 0.8382 - val_loss: 0.3717
Epoch 19/20
51/51 ━━━━━━━━━━━━━━━━━━━━ 27s 519ms/step - accuracy: 0.8348 - loss: 0.4188 - val_accuracy: 0.8555 - val_loss: 0.5269
Epoch 20/20
51/51 ━━━━━━━━━━━━━━━━━━━━ 26s 519ms/step - accuracy: 0.8398 - loss: 0.3877 - val_accuracy: 0.9075 - val_loss: 0.3016
```

Figura 5.1: Evoluția procesului de antrenare pe parcursul epocilor

Se observă o creștere progresivă a acurateții și o scădere a funcției de pierdere, indicând faptul că modelul învață tipare relevante din datele de intrare.

6. Evaluarea performanței

6.1 Evaluarea clasificării multi-clasă

Evaluarea performanței modelului este realizată pe setul de test, utilizând metrici standard precum precision, recall și F1-score.

```
6/6 ━━━━━━━━━━━━━━━━━━━━ 2s 252ms/step
```

	precision	recall	f1-score	support
Bronchiectasis	0.00	0.00	0.00	3
Bronchiolitis	0.00	0.00	0.00	2
COPD	0.95	0.95	0.95	133
Healthy	0.88	0.88	0.88	16
Pneumonia	0.56	0.88	0.68	16
URTI	0.00	0.00	0.00	3
accuracy			0.89	173
macro avg	0.40	0.45	0.42	173
weighted avg	0.87	0.89	0.88	173

Figura 6.1: Raportul de clasificare pentru problema multi-clasă

Rezultatele obținute arată performanțe foarte bune pentru clasele bine reprezentate în setul de date, precum *COPD* și *Healthy*, unde modelul obține valori ridicate ale preciziei și recall-ului. În schimb, clasele rare obțin scoruri scăzute, ceea ce este explicabil prin numărul insuficient de exemple disponibile pentru antrenare. Clasa *Pneumonia* prezintă un recall ridicat, dar o precizie mai redusă, indicând faptul că modelul identifică majoritatea cazurilor reale, însă realizează și unele confuzii. Acuratețea globală de aproximativ 89 % confirmă o performanță bună, în contextul dezechilibrului setului de date.

6.2 Analiza curbelor de învățare

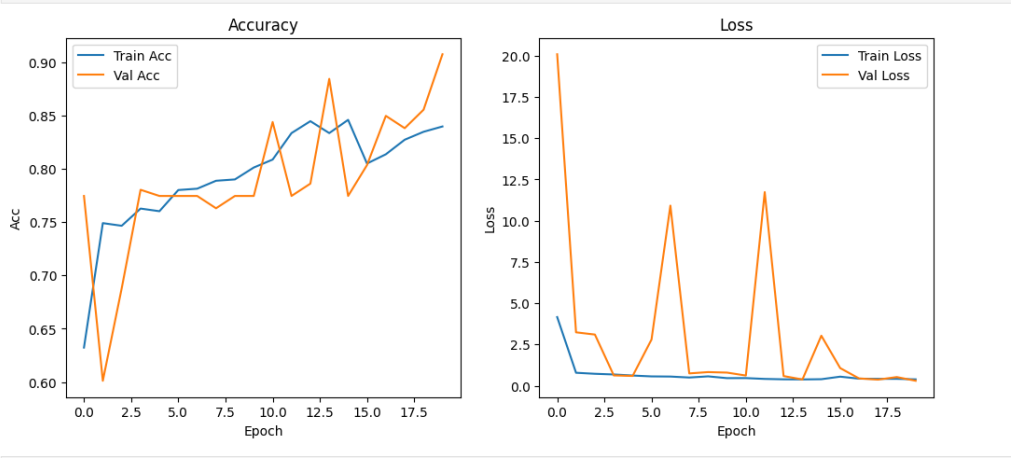

Figura 6.2: Curbele de acuratețe și funcția de pierdere

Curbele de învățare arată că acuratețea pe setul de antrenare crește constant, indicând faptul că modelul învață progresiv tipare relevante din date. Acuratețea pe setul de validare are o evoluție similară și rămâne, în general, apropiată de cea de antrenare, ceea ce sugerează o bună capacitate de generalizare și absența fenomenului de overfitting.

În cazul funcției de pierdere, se observă o scădere treptată a loss-ului pe setul de antrenare, în timp ce loss-ul de validare prezintă spike-uri izolate, vizibile sub forma unor creșteri bruște ale valorii loss-ului în anumite epoci. Aceste spike-uri apar atunci când, într-o epocă, modelul întâlnește exemple mai dificile sau clase slab reprezentate și sunt explicabile prin dezechilibrul setului de date și numărul redus de exemple pentru anumite patologii. Prezența acestor spike-uri, în lipsa unei creșteri constante a loss-ului de validare, nu indică supraînvățare.

6.3 Detecția anomaliilor

Pentru a depăși limitările clasificării multi-clasă, problema a fost reformulată ca detecție de anomalii, considerând clasa Healthy drept normală și toate celelalte patologii drept anormale.

```
Folosim clasa normală: Healthy
              precision    recall  f1-score   support

      Normal       0.88      0.88      0.88        16
    Abnormal       0.99      0.99      0.99       157

    accuracy                           0.98       173
   macro avg       0.93      0.93      0.93       173
weighted avg       0.98      0.98      0.98       173
```

Figura 6.3: Evaluarea detecției de anomalii (Normal vs Abnormal)

Această abordare conduce la o acuratețe de aproximativ 98 %, indicând faptul că modelul reușește să diferențieze corect, în majoritatea cazurilor, între sunetele respiratorii normale și cele anormale. Recall-ul foarte ridicat pentru clasa *Abnormal* arată că modelul identifică aproape toate cazurile patologice, ceea ce este deosebit de important într-un context medical, unde ratarea unor cazuri anormale poate avea consecințe semnificative. Aceste rezultate demonstrează că formularea problemei ca detecție de anomalii este mai robustă și mai potrivită pentru setul de date utilizat.

7. Concluzii

7.1 Concluzii generale

În cadrul acestui proiect a fost dezvoltat un sistem bazat pe cunoaștere pentru analiza automată a sunetelor respiratorii, utilizând rețele neuronale convoluționale și spectrograme Log-Mel. Rezultatele obținute arată că modelul este capabil să distingă eficient între starea normală și cea patologică.

Clasificarea multi-clasă este limitată de dezechilibrul setului de date, însă reformularea problemei ca detecție de anomalii conduce la performanțe semnificativ mai bune.

7.2 Direcții viitoare de dezvoltare

Direcțiile viitoare de dezvoltare includ:

- augmentarea datelor audio pentru creșterea numărului de exemple;
- utilizarea unor metode de echilibrare a claselor;
- explorarea unor modele dedicate detecției de anomalii;
- integrarea sistemului într-o aplicație de suport decizional medical.

Bibliografie

[1] Wikipedia contributors, *Spectrogram*, Wikipedia, The Free Encyclopedia, 2025. Disponibil online: `https://en.wikipedia.org/wiki/Spectrogram`

[2] Wikipedia contributors, *Mel scale*, Wikipedia, The Free Encyclopedia, 2025. Disponibil online: `https://en.wikipedia.org/wiki/Mel_scale`

[3] K. Doshi, *Audio Deep Learning Made Simple: Why Mel Spectrograms Perform Better*, Medium, 2021. Disponibil online: `https://medium.com/data-science/audio-deep-learning-made-simple-part-2-why-mel-spectrograms-perform-better-aad889a93505`

[4] Scott Duda, *Urban Environmental Audio Classification Using Mel Spectrograms*, Medium, 2020. Disponibil online: `https://scottmduda.medium.com/urban-environmental-audio-classification-using-mel-spectrograms-706ee6f8dcc1`

[5] Brian McFee et al., *Librosa: Audio and Music Signal Analysis in Python*, Documentație oficială, 2025. Disponibil online: `https://librosa.org/doc/latest/index.html`

[6] TensorFlow Team, *TensorFlow Documentation – Convolutional Neural Networks*, Google, 2025. Disponibil online: `https://www.tensorflow.org/tutorials/images/cnn`

[7] GeeksforGeeks, *Audio Classification using Spectrograms*, 2025. Disponibil online: `https://www.geeksforgeeks.org/nlp/audio-classification-using-spectrograms/`

[8] crlandsc, *Music Genre Classification Using Convolutional Neural Networks*, GitHub Repository. Disponibil online: `https://github.com/crlandsc/Music-Genre-Classification-Using-Convolutional-Neural-Networks`

[9] Materiale de curs furnizate în cadrul disciplinei *Sisteme bazate pe cunoaștere*, Universitatea Tehnică din Cluj-Napoca, An universitar 2025–2026.