Universitat Politècnica de Catalunya

Facultat d'Informàtica de Barcelona

Pràctica 4,
Reconeixement de comandes de veu

Tractament de la Veu i el Diàleg

Autors:

Zhihao Chen; Zhiqian Zhou

31 de desembre de 2024

Índex

1 Introducció

Aquest projecte se centra en el reconeixement de comandes de veu, una tasca clau en l'àmbit del processament de senyals i l'aprenentatge profund. L'objectiu és construir models capaços de classificar gravacions de veu en un conjunt de comandes predefinides. Aquestes comandes inclouen paraules com ara "up", "left", "bird", "on" i "five", entre altres.

El conjunt de dades proporcionat conté gravacions d'àudio etiquetades amb els respectius comandaments, que serveixen com a base per entrenar i avaluar els models. El propòsit principal de la pràctica és explorar diferents tècniques d'aprenentatge automàtic per resoldre aquesta tasca, implementant models que utilitzin les estratègies apreses durant el curs.

2 Metodologia

La metodologia s'ha dissenyat en dues fases principals:

1. **Fase 1: Avaluació d'arquitectures de models**

   Inicialment, es van explorar i implementar diverses arquitectures per al modelatge de dades, incloent-hi xarxes recurrents (RNN, GRU, LSTM), xarxes convolucionals (CNN) i transformers. L'objectiu d'aquesta fase és identificar quina arquitectura ofereix millor rendiment en el conjunt de dades proporcionat.

   Per garantir estabilitat i evitar sobreajustament, es van utilitzar tècniques de normalització (normalització Batch o de capa) i regularització (*dropout* i *early stopping*).

2. **Fase 2: Experimentació amb tipus d'inputs**

   Una vegada seleccionada la millor arquitectura en la fase 1, es va provar diferents representacions dels inputs. Això inclou espectrograma lineal, espectrograma MEL i MFCC. Per a cada tipus d'input, es va entrenar i avaluar el model seleccionat per determinar la configuració òptima.

3 Preprocessament bàsic de dades

En el procés de preprocessament bàsic, primer es defineix el *path* on es troben les dades. Amb la funció *audio_dataset_from_directory* es carreguen els arxius d'àudio, es posen en batchs de mida 64 i es divideixen en conjunts d'entrenament i validació amb una proporció del 80%-20% (el conjunt de validació se separarà per test i validació). També s'assegura que les seqüències d'àudio tinguin una longitud consistent de 16.000 mostres, equivalent a un segon de durada per freqüència de mostreig de 16 kHz. Després, s'aplica la funció *squeeze* per eliminar la dimensió extra de canals de les mostres d'àudio. Finalment, el conjunt de validació es divideix en dos subconjunts: un de validació i un altre de test, utilitzant la funció *shard*.

A continuació, es transforma els waveforms en espectrogrames mitjançant la funció *get_spectrogram*, que aplica una STFT amb una longitud de finestra de 255 mostres i un pas de 128 mostres. Això genera una representació en freqüència del senyal. Es calcula la magnitud de la STFT i s'afegeix una dimensió de canals per preparar els espectrogrames com a entrada dels models. Els conjunts de dades d'entrenament, validació i test es converteixen en conjunts d'espectrogrames mitjançant la funció *make_spec_ds*. Finalment, es fa una optimització dels conjunts amb memòria cau (*cache*), barrejant les dades d'entrenament (*shuffle*) i precarregant lotes (*prefetch*) per millorar l'eficiència durant l'entrenament del model.

4 Definició dels models

4.1 Model Baseline

El model baseline utilitzat és una xarxa neuronal convolucional senzilla dissenyada per classificar espectrogrames d'àudio en diferents comandes de veu. La xarxa consta d'una capa de redimensionament, una capa de normalització, una capa convolucional amb activació ReLU, una capa de max-pooling, una capa densa amb 64 unitats i finalment una capa de sortida amb el nombre de classes corresponent. El model es compila amb l'optimitzador Adam i es fa servir la funció de pèrdua *SparseCategoricalCrossentropy* per a la classificació múltiple.

El rendiment del model en el conjunt de test és relativament baix, amb un *accuracy* de 0.6911, mentre que en el conjunt d'entrenament és 0.7224, suggerint un possible sobreajustament. La seva arquitectura és molt senzilla, amb només una capa convolucional i una capa densa, el que limita la seva capacitat per extreure característiques complexes dels espectrogrames. A més, el model compta amb un total de 463199 paràmetres entrenables, una quantitat modesta per a un problema tan exigent com el reconeixement de veu. També cal destacar que no s'utilitzen tècniques avançades de preprocessament. Els inputs que usa són espectrogrames lineals amb un *frame length* de 255 i un *frame step* de 128.

En resum, el model baseline serveix com a punt de partida, però presenta diverses limitacions que es poden superar mitjançant models més profunds i tècniques de preprocessament més sofisticades.

4.2 Ajustament del nombre d'èpoques

La primera millora realitzada és l'ajustament del nombre d'èpoques. En observar que el model baseline mostrava una tendència d'augment del *accuracy* tant en la partició d'entrenament com en la partició de validació a mesura que avançaven les èpoques, es decideix incrementar el nombre d'èpoques de 5 a 50. Tot i així, es considera que aquest nombre d'èpoques podia ser excessiu, fet que podria provocar un efecte de *overfitting*. Per evitar-ho, s'aplica una tècnica de regularització anomenada *early stopping*, amb una paciència de 5, de manera que el model s'aturaria si el rendiment de validació no millorava després de 5 èpoques consecutives.

Després de 15 èpoques d'entrenament, s'observa que l'*accuracy* del model baseline augmenta considerablement, assolint un *accuracy* de 0.8497 en entrenament i 0.7500 en validació. Com era d'esperar, l'augment en el nombre d'èpoques provoca un lleuger augment del risc d'*overfitting*, però també dona com a resultat una millora significativa en el rendiment de validació. La millora relativa en el *accuracy* de validació és d'aproximadament 7.62%.

4.3 Xarxes Recurrents

El segon experiment se centra a provar noves arquitectures de xarxes neuronals profundes, concretament les xarxes neuronals recurrents (RNN). Per tal d'estalviar recursos i temps d'entrenament, es decideix provar directament les arquitectures més avançades, com les LSTM (Long Short-Term Memory) i GRU (Gated Recurrent Units), sense utilitzar una RNN bàsica. Aquestes arquitectures són conegudes per la seva capacitat d'aprendre seqüències de dades temporals i poden millorar el rendiment en tasques com el reconeixement de veu.

En aquest experiment, es manté constant el nombre de capes recurrents per a totes les arquitectures, provant tant una configuració amb 5 com amb 7 capes. S'utilitza una capa de *Reshape* per adaptar la forma dels espectrogrames a les capes recurrents. A més, s'aplica una capa normalització a l'entrada als blocs recurrents i una capa de *Dropout* per evitar el sobreajustament.

Una característica important d'aquest experiment és la prova de models unidireccionals i bidireccionals. Els

models bidireccionals permeten que la xarxa tingui accés a la informació dels dos extrems de la seqüència d'entrada, oferint així un context global que pot ser essencial per a tasques de reconeixement de veu, on el significat d'un comandament pot dependre tant de les paraules anteriors com de les següents.

L'aprenentatge dels models experimentats tant en el conjunt de dades d'entrenament com en el de validació es presenta a continuació:

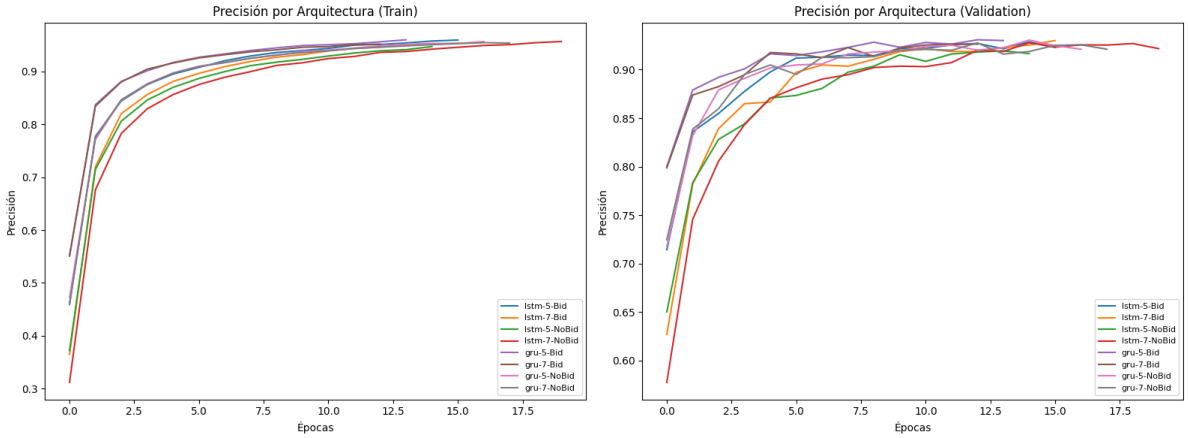

Figura 1: Aprenentatge dels models RNN

Tal com es pot observar en la figura, l'aprenentatge dels models RNN ha estat consistent i estable al conjunt d'entrenament, amb un augment significatiu en el rendiment de les prediccions. Tots els models experimentats han aconseguit un *accuracy* superior a 0.9 en tots dos conjunts de dades, demostrant la bona capacitat de les xarxes recurrents amb 5 o 7 capes en la tasca de reconeixement de comandes de veu. Així mateix, el rendiment positiu sembla ser independent tant de l'arquitectura utilitzada (LSTM o GRU) com de la tècnica bidireccional, tot i que l'ús d'arquitectures bidireccionals incrementa notablement el temps d'entrenament.

A continuació es mostra la taula amb els resultats obtinguts per a cada model en el conjunt d'entrenament i validació, la qual cosa demostra que hi existeix una lleugera sobreestimació, però no gaire greu gràcies a les capes de *Dropout* definides:

Model	Capes	Bid.	Train	Val
LSTM	5 Capes	Sí	0.959754	0.922958
		No	0.947318	0.916615
	7 Capes	Sí	0.953536	0.930074
		No	0.956935	0.921720

Model	Capes	Bid.	Train	Val
GRU	5 Capes	Sí	0.960160	0.930074
		No	0.956742	0.921102
	7 Capes	Sí	0.951952	0.926361
		No	0.953999	0.921102

Figura 2: Accuracy d'entrenament i validació per a diferents models RNN

D'acord amb els resultats obtinguts, el model GRU bidireccional amb 5 capes, que té fins a 1.5M paràmetres entrenables, és el que presenta el millor *accuracy* en el conjunt de validació, tenint en compte alhora la complexitat del model. Aquest assoleix un valor de 0.9301. Això representa una millora d'aproximadament un 12.41% en comparació amb l'experiment anterior, destacant la capacitat d'aquest model per generalitzar millor a dades no vistes durant l'entrenament.

4.4 Xarxes Convolucionals

En aquest experiment, es prova tres models convolucionals de diferents complexitats amb l'objectiu d'augmentar el rendiment del model de base en la tasca de reconeixement de veu. La diferència principal entre aquests models és la profunditat de les xarxes i la inclusió de tècniques de regularització com el *Dropout*.

El primer model, de complexitat bàsica, consta d'una sola capa convolucional seguida d'una capa de *max-pooling* i una capa densa per a la classificació. Aquest model és senzill, però la seva capacitat d'aprendre característiques complexes és limitada. En canvi, els models de complexitat intermediària i avançada afegeixen més capes convolucionals i de *max-pooling*, augmentant la capacitat de la xarxa per capturar característiques més abstractes dels espectrogrames. A més, s'afegeixen capes de *Dropout* després de cada bloc convolucional i abans de la capa de classificació per evitar l'*overfitting*, un problema comú en xarxes amb una gran profunditat.

S'ha dissenyat aquest experiment amb la intenció d'explorar l'impacte de l'augment de la complexitat de les xarxes convolucionals, així com el seu efecte en la precisió i la capacitat de generalització del model. A mesura que la profunditat de les xarxes augmentava, s'ha observat que la regularització era essencial per evitar que el model s'adaptés massa als detalls específics del conjunt d'entrenament, millorant així la seva capacitat de generalitzar a noves dades:

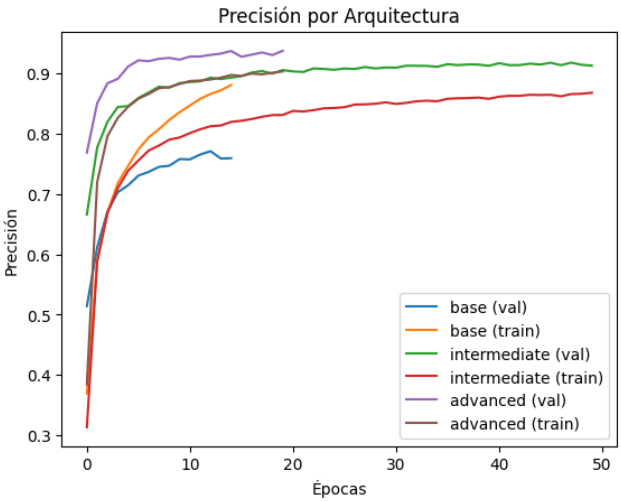

Figura 3: Aprenentatge dels models CNN

Nivell	Train	Val
Base	0.881175	0.759592
Intermediate	0.868236	0.913212
Advanced	0.903596	0.937809

Figura 4: Accuracy d'entrenament i validació per a diferents models CNN

Tal com es pot observar, el model bàsic de CNN només aconsegueix un *accuracy* de 0.76 en la validació, presentant també una greu sobreestimació. En canvi, els models intermedi i avançat, que incorporen més capes convolucionals i *Dropout*, mostren una millora considerable. En aquests models, la línia d'aprenentatge en el conjunt d'entrenament es manté sempre per sota de la línia de validació, evitant així el sobreajustament. El model avançat, amb més complexitat, aconsegueix un *accuracy* de 0.9378 en la validació, demostrant la seva capacitat de reconeixement de comandes de veu. Aquest resultat és comparable amb el de les xarxes recurrents, però en aquest cas, el model CNN és capaç de capturar contextos locals acumulats gràcies a l'augment de la profunditat i aprofitar-los per reconèixer correctament la veu.

4.5 Transformers

En els experiments amb arquitectures basades en Transformers, s'ha dut a terme un petit *benchmark* per explorar els hiperparàmetres òptims. Entre aquests es troben la dimensió dels *embeddings*, el nombre de blocs de Transformers, el nombre de caps d'atenció i el nombre de neurones en les xarxes *feedforward* internes.

L'entrenament d'aquests models resulta ser significativament lent, amb temps de fins a 10 minuts per època. Per accelerar el procés, s'han redimensionat les entrades als blocs de Transformers en "patches", reduint la mida dels espectrogrames de 32x32 a 4x4. Tot i que aquesta tècnica permetia que els experiments fossin viables amb els recursos disponibles, s'ha comportat una pèrdua de precisió en els resultats.

Com era d'esperar, els resultats no són tan bons com els obtinguts amb altres arquitectures. Els models Transformers aconsegueixen un *accuracy* d'aproximadament 0.8 en el conjunt de validació, tot i que el model més complex té un nombre de paràmetres entrenables que arribava als 2M. A més, la variabilitat en el rendiment entre les diferents configuracions d'hiperparàmetres indica que aquests models són menys estables i requereixen una cerca d'hiperparàmetres més extensa per assolir un bon rendiment. Això posa de manifest la necessitat de més recursos computacionals per explorar adequadament les capacitats dels Transformers en aquesta tasca.

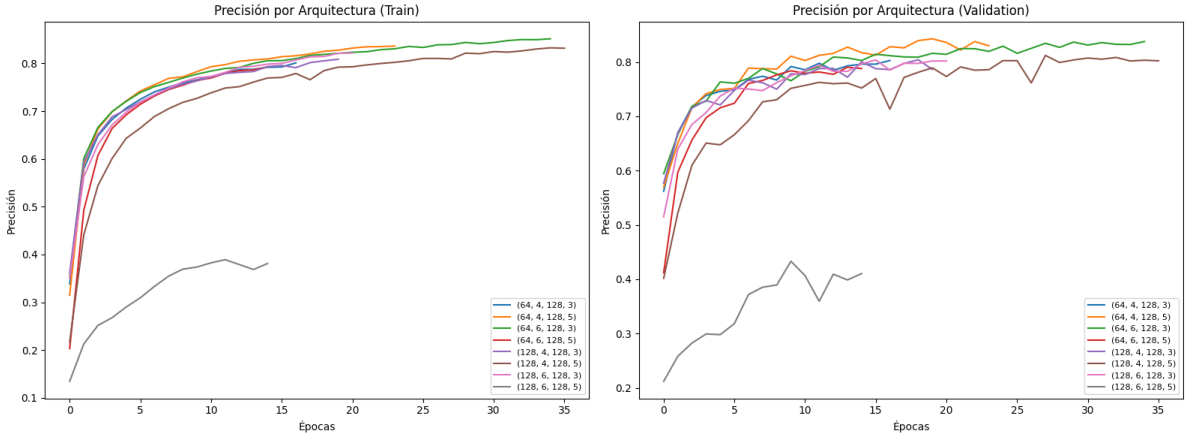

Figura 5: Aprenentatge dels models Transformers.

Els resultats detallats de precisió per a diferents configuracions d'hiperparàmetres es mostren en les taules següents. Cada configuració es representa com una tupla que indica la dimensió dels *embeddings*, el nombre de caps d'atenció, el nombre de neurones en les xarxes *feedforward* i el nombre de blocs de Transformers.

Configuració	**(64, 4, 128, 3)**	**(64, 4, 128, 5)**	**(64, 6, 128, 3)**	**(64, 6, 128, 5)**	**(128, 4, 128, 3)**	**(128, 4, 128, 5)**	**(128, 6, 128, 3)**	**(128, 6, 128, 5)**
Train	0.801089	0.836159	0.851416	0.792631	0.808544	0.831718	0.821753	0.381271
Val	0.802908	0.829981	0.837717	0.788057	0.786510	0.802290	0.801980	0.410427

Taula 1: Accuracy per a models Transformers amb diferents configuracions.

5 Preprocessament avançat de dades

En aquesta fase, s'ha implementat una funció generalitzada per a la generació d'espectrogrames a partir de les formes d'ona dels àudios. Aquesta funció pren com a entrada una forma d'ona i retorna l'espectrograma especificat, que pot ser lineal, MEL o MFCC. Els paràmetres ajustables inclouen la longitud de la finestra, el pas entre finestres, i, en el cas dels espectrogrames MEL i MFCC, el nombre de bandes MEL i coeficients MFCC. Els punts que cal tenir en compte són:

- Es pot aplicar una escala logarítmica per a millor interpretabilitat, ja que ajuda a reflectir millor com l'oïda humana percep el so.
- Per aquesta tasca, s'han ajustat els paràmetres de manera que l'espectrograma lineal resultant sigui quadrada (mateix nombre de finestres temporals i freqüències), pel fet que aquest format ofereix avantatges importants per a xarxes convolucionals (el qual s'està fent servir), com la simetria estructural i la consistència en el processament espacial.
- Els diferents tipus d'espectrogrames s'obtenen de manera jeràrquica, començant per l'espectrograma lineal, que es genera mitjançant la transformada curta de Fourier (STFT) i l'extracció de la magnitud de les freqüències. A partir d'aquest, es construeix l'espectrograma MEL utilitzant una matriu de pesatge que

ajusta les freqüències a una escala perceptiva humana. Finalment, els MFCC s'obtenen aplicant la transformada de cosinus discreta (DCT) sobre les bandes MEL per retenir els coeficients més rellevants. Aquesta estructura jeràrquica permet optimitzar el procés experimental: primer es determina el millor espectrograma lineal, després el millor espectrograma MEL basat en aquest, i finalment el millor MFCC. Això redueix significativament el nombre d'experiments necessaris.

5.1 Optimització de l'espectrograma lineal

L'objectiu és trobar els millors valors per a *frame_length*, *frame_step* i *use_db*. Els valors de *frame_length* que es provaran són 256, 512 i 1024, cadascun amb els seus respectius *frame_step* de 128, 256 i 512, de manera que l'espectrograma sigui tan quadrat com sigui possible. Un cop trobada la millor combinació, es provarà si utilitzar una escala logarítmica (*use_db*) millora el rendiment del model.

5.1.1 Frame-length i Frame-step

S'han provat els valors de *frame_length* i *frame_step* (256, 128), (512, 256), (1024, 512), i s'han obtingut les dimensions dels espectrogrames (124, 129, 1), (259, 257, 1) i (500, 513, 1), respectivament:

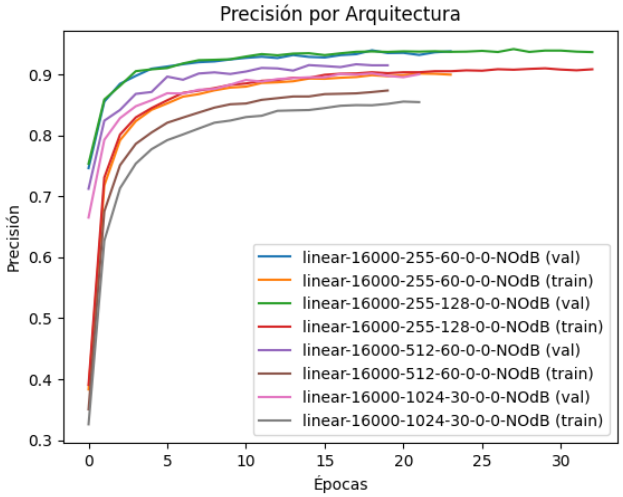

Figura 6: Aprenentatge per a models amb diferents configuracions d'espectrograma lineal (provant mides i nombres de finestres)

Configuració	Train	Val
(255,128)	0.908366	0.936572
(512-60)	0.873547	0.914913
(1024-30)	0.854544	0.900217

Figura 7: Accuracy per a models amb diferents configuracions d'espectrograma lineal (provant mides i nombres de finestres).

Després d'entrenar el model amb diferents configuracions, s'ha vist que la combinació 'frame_length = 256' i 'frame_step = 128' produeix els millors resultats. Això indica que augmentar la dimensió de l'espectrograma (amb més resolució freqüencial i temporal) no sempre millora el rendiment del model.

Un dels factors que podria explicar aquest resultat és que abans d'entrar a la CNN es realitza un *resampling*, que redueix la resolució i fa que espectrogrames més grans no aportin un avantatge significatiu. A més, espectrogrames més grans poden incloure informació freqüencial addicional, però aquesta informació pot ser redundant o irrellevant per a la tasca.

5.1.2 L'ús de dB

Un cop determinada la millor configuració per a *frame_length* i *frame_step*, el següent pas és provar si utilitzar l'escala en dB (*use_db*) millora el rendiment del model, ja que aquesta transformació imita la percepció humana

del so.

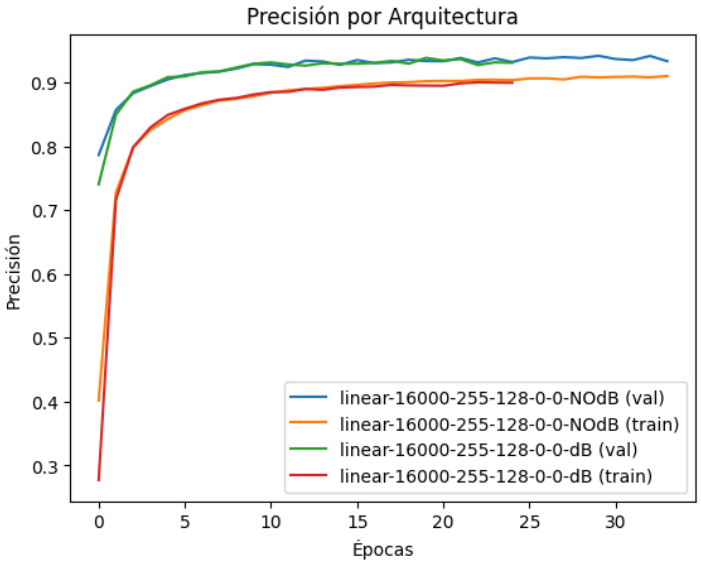

Figura 8: Aprenentatge per a models amb diferents configuracions d'espectrograma lineal (provant l'ús de les unitats dB).

Configuració	Train	Val
NOdB	0.910027	0.933323
dB	0.899598	0.930848

Figura 9: Accuracy per a models amb diferents configuracions d'espectrograma lineal (provant l'ús de les unitats dB).

Els resultats mostren que no utilitzar l'escala en dB ofereix lleugerament millors resultats, amb una precisió de 0.9100 en entrenament i 0.9333 en validació, en comparació amb els valors obtinguts amb l'escala en dB (0.8996 en entrenament i 0.9308 en validació). Tot i que la diferència és petita, aquesta observació suggereix que, en aquest cas, la transformació logarítmica no aporta una millora significativa.

5.2 Optimització de l'espectrograma MEL

L'objectiu és trobar el millor valor per a *num_mel_bins* mantenint els paràmetres optimitzats de *frame_length*, *frame_step* i *use_db* obtinguts prèviament amb l'espectrograma lineal. Els valors que es provaran per a *num_mel_bins* són 80, 128 i 256, ja que aquest paràmetre afecta la resolució freqüencial de l'espectrograma MEL.

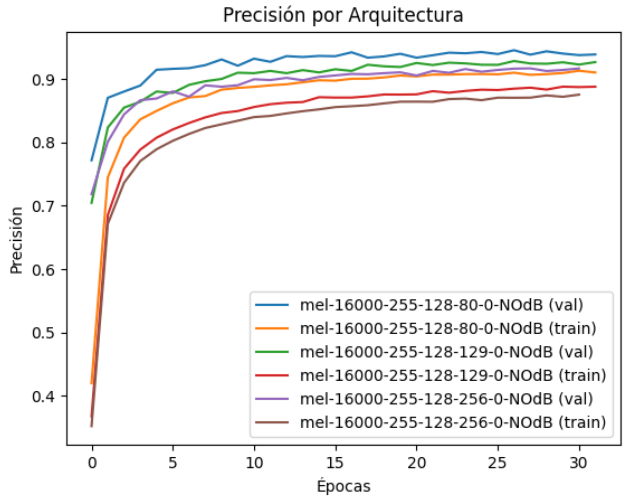

Figura 10: Aprenentatge per a models amb diferents configuracions d'espectrograma mel (provant nombres de filtres mel).

Configuració	Train	Val
80	0.910297	0.938738
129	0.887837	0.926671
256	0.875362	0.916615

Figura 11: Comparació de resultats per diferents valors de `num_mel_bins` en espectrogrames MEL.

Els resultats mostren que l'espectrograma MEL amb 80 `num_mel_bins` aconsegueix un millor rendiment que

amb un nombre més gran de `num_mel_bins`. Això possiblement és degut al fet que utilitzar un nombre més alt de bins pot causar que la xarxa sigui més sensible al soroll i menys capaç d'aprendre les relacions fonamentals. Amb 80 bins, però, l'espectrograma MEL aconsegueix una representació més compacta i robusta de la informació de l'àudio. A més, l'espectrograma MEL supera l'espectrograma lineal perquè la representació MEL és més semblant a la percepció auditiva humana, per tant, permet una extracció més eficaç de característiques clau.

5.3 Optimització dels MFCCs

L'objectiu és trobar el millor valor per a *num_mfccs* mantenint els altres paràmetres optimitzats a l'experiment anterior. Els valors a provar són 10, 13, 20 i 30 que afecten la quantitat de coeficients utilitzats com a entrada al model. Més coeficients preservaran més informació, però també augmentaran la complexitat computacional.

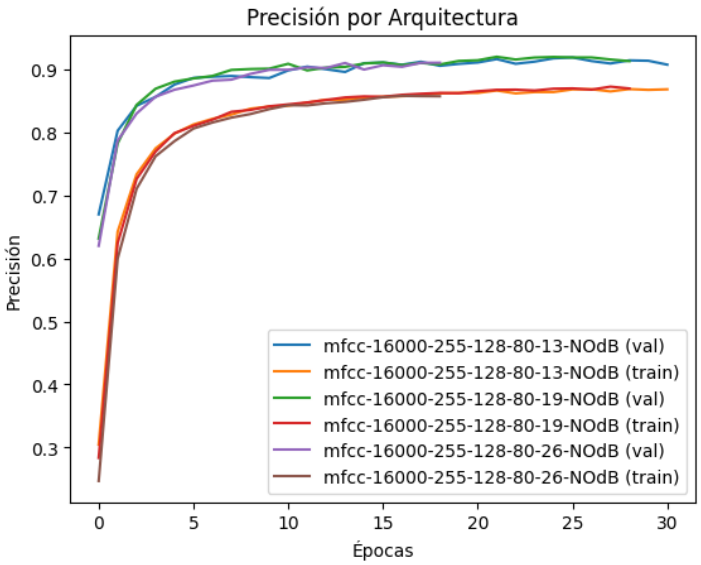

Figura 12: Aprenentatge per a models amb diferents configuracions d'espectrograma mfcc (provant nombres de components).

Configuració	Train	Val
13	0.868371	0.907488
19	0.869684	0.913057
26	0.857325	0.910582

Figura 13: Resultats de l'experimentació amb diferents valors de `num_mfccs` (13, 19 i 26) per espectrograma MFCC.

Tot i que els models amb MFCC basats en el millor espectrograma mel anterior han obtingut bons resultats, cap d'ells supera el model amb mel, obtenint una precisió de validació aproximadament 0.02 punts inferior. Això es podria deure al fet que els coeficients MFCC representen una versió més comprimida de l'espectrograma mel, cosa que pot comportar una pèrdua d'informació important per a la tasca de classificació. En canvi, l'espectrograma de mel, amb més informació de les freqüències, sembla que conserva més detalls i millora el rendiment en aquest cas.

6 Altres millores realitzades

6.1 Xarxes Convolucionals Profundes

Els experiments amb xarxes convolucionals han demostrat que aquestes arquitectures són eficaces en el reconeixement de veu. En particular, s'ha observat que l'augment de la complexitat del model millora el rendiment, amb l'arquitectura avançada assolint un *accuracy* del 0.94 en validació. Tanmateix, la regularització afegida en aquest model s'ha provocat un lleuger *underfitting*.

Per abordar aquesta limitació, es proposa augmentar encara més la complexitat del model, fins a arribar a uns 9 milions de paràmetres entrenables. Aquest canvi permetria un millor ajustament a les dades d'entrenament,

mentre es manté l'eficàcia de les tècniques de regularització per evitar l'*overfitting*. Amb aquest enfocament, es millora l'*accuracy* en validació fins al 0.96 i aconseguir una convergència equilibrada entre entrenament i validació, eliminant tant l'*underfitting* com l'*overfitting*.

6.2 Combinació de Xarxes Recurrents i Convolucionals

En el processament de veu amb *deep learning*, és habitual combinar xarxes recurrents (RNN) i convolucionals (CNN), ja que cadascuna aporta avantatges únics: la CNN capten el context local, mentre que les RNN modelen la informació seqüencial.

Aquest enfocament combinat podria assolir resultats similars als d'una xarxa convolucional profunda, però amb una arquitectura encara més complexa, amb uns 14 milions de paràmetres. Tanmateix, un desavantatge important és la dificultat d'entrenar aquesta combinació, ja que les RNN tenen un cost computacional elevat i no són tan fàcilment de ser paral·lelitzades com la CNN.

6.3 Data Augmentation

Quan les dades disponibles són limitades, el *data augmentation* és una estratègia clau per millorar el rendiment dels models de *deep learning*. Partint de la base que l'*accuracy* de les arquitectures avançades sembla estabilitzar-se al voltant del 0.96 en validació, s'ha decidit explorar aquesta tècnica.

S'apliquen mètodes de *data augmentation* basats en *time warping* i *masking*, seguint els enfocaments vistos a classe. No obstant això, els resultats són decebedors, amb una baixada de l'*accuracy* fins al 0.92 en validació. Aquesta disminució probablement es deu a una implementació subòptima, amb paràmetres mal ajustats, com el nombre de finestres *masquerajades* o la longitud de l'espectrograma transformat.

Amb una exploració més exhaustiva dels hiperparàmetres d'aquestes tècniques de *data augmentation*, seria possible obtenir beneficis més significatius i superar els límits observats fins ara.

7 Normalització i Regularització

En aquesta pràctica es demanava explícitament provar les tècniques de normalització i regularització. Tot i això, no s'ha obert una secció específica per analitzar-ne directament els impactes, ja que les arquitectures implementades ja les incorporen de manera intrínseca.

Pel que fa a la normalització, aquesta ha ajudat els models a estabilitzar l'aprenentatge i accelerar la convergència. Des del model *baseline*, es va utilitzar una capa de normalització abans de les capes convolucionals, la qual cosa ja assegurava certa estabilitat durant l'entrenament. Els models més avançats van seguir aquesta mateixa línia, i s'hi van afegir capes de *Batch Normalization* després de cada capa convolucional, contribuint a la millora del rendiment sense evidenciar dificultats en la convergència. En relació amb la regularització, també està integrada en les arquitectures analitzades. Al model *baseline*, que inicialment no en disposava, es va observar un marcat efecte de sobreajustament (*overfitting*). Als models posteriors, la inclusió de capes de *Dropout* va permetre reduir aquest problema, arribant a evitar-lo completament. De fet, en algunes configuracions de xarxes convolucionals avançades, la regularització va ser tan efectiva que va conduir a un lleuger efecte de *underfitting*, que es va solucionar augmentant la complexitat del model.

Una altra raó per la qual no s'han provat explícitament els models sense aquestes tècniques és que arquitectures com els Transformers ja integren capes de normalització i regularització per defecte, com ara *Layer Normalization*

o *Dropout*. Atès que l'objectiu principal d'aquesta pràctica era comparar diferents models en la tasca de reconeixement de veu, es va optar per mantenir aquestes tècniques activades en totes les arquitectures per garantir una comparació justa i coherent entre elles.

8 Anàlisi de resultats

En el model baseline, la matriu de confusió havia mostrat que la major part dels errors es produeixen entre comandes amb pronunciacions similars, com ara "go" i "no". A mesura que l'arquitectura dels models ha evolucionat, s'ha aconseguit reduir aquesta confusió entre comandes, tot i que encara es mantenen alguns errors en determinades comandes, però en menor quantitat. Aquestes confusions podrien ser resoltes mitjançant una neteja més exhaustiva de les dades, una tasca que no es va dur a terme completament durant aquesta pràctica.

D'altra banda, l'única confusió que supera els 6 errors en la matriu de confusió del millor model obtingut (CNN profunda) es produeix entre les comandes "off" i "up", amb un total de 13 errors. Aquesta situació suggereix que una possible millora seria reduir la mida de la finestra per millorar la capacitat del model per distingir entre aquestes dues comandes, que tenen una durada de pronunciació relativament curta. Tot i això, en un altre model amb un rendiment similar (RNN + CNN), no s'ha observat aquesta confusió, tot i que aquest model ha mostrat un nombre més alt d'errors menors en general.

9 Conclusió

Aquest projecte ha estat una aplicació pràctica de les tècniques d'aprenentatge automàtic estudiades durant el curs, centrant-se en el reconeixement de veu mitjançant diverses arquitectures i estratègies. A la fase 1, es van explorar models com les xarxes recurrents (GRU, LSTM), xarxes convolucionals (CNN) i transformers. Els resultats van indicar que les CNN són les més adequades per aquesta tasca, ja que aconsegueixen un bon equilibri entre precisió i temps d'entrenament. Paral·lelament, es van aplicar tècniques de normalització i regularització per garantir l'estabilitat dels models i prevenir el sobreajustament. A la fase 2, es van comparar diferents representacions d'entrada com l'espectrograma lineal, l'espectrograma MEL i els coeficients MFCC. Els resultats van destacar que els espectrogrames MEL amb 80 bins van oferir el millor rendiment, subratllant la importància de paràmetres com la mida i el pas de la finestra, l'ús d'escala logarítmica, i la resolució espectral en la qualitat del model.

Tot i l'èxit aconseguit, els experiments van mostrar que el rendiment dels models profunds queda limitat a un *accuracy* màxim d'aproximadament 0.96 en la validació. Això suggereix que, més enllà de continuar augmentant la complexitat dels models, caldria prioritzar el preprocessament de dades. La manca de neteja i augment de dades adequats probablement ha influït en les limitacions observades. També es va detectar potencial en l'ús d'espectrogrames no quadrats, com ara configuracions amb mida de finestra 255 i pas 60, que van mostrar un rendiment prometedor. Finalment, els intents de *data augmentation* no van permetre superar el coll d'ampolla del rendiment. Això indica que, o bé les dades actuals són insuficients, o bé els paràmetres d'augment no van ser òptims. De cara a futurs treballs, seria recomanable:

- Realitzar un preprocessament de dades més profund per millorar la qualitat del conjunt d'entrenament.
- Investigar a fons l'ús d'espectrogrames no quadrats per optimitzar la representació dels senyals.
- Explorar models pre-entrenats que podrien ajudar a superar les limitacions derivades de la manca de dades.