STUDIUL ȘI EXPERIMENTAREA RECUNOȘTERII AUTOMATE A VORBIRII ÎN LIMBA ROMÂNĂ FOLOSIND DEEPSPEECH

ASRSV 14.12.2021

HORNAI VLAD

Cuprins

Motivatie

State of the art

Fundamentare Teoretică

Componentele unui sistem de recunoaștere a vorbirii

- Componenta acustică
- Modelul acustic
- Modelul lingvistic
- Decodorul

Arhitectura DeepSpeech

Procesarea datelor

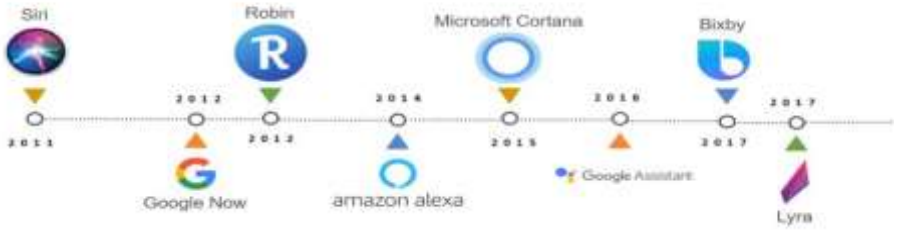
Rezultate experimentale

Concluzii

Motivație

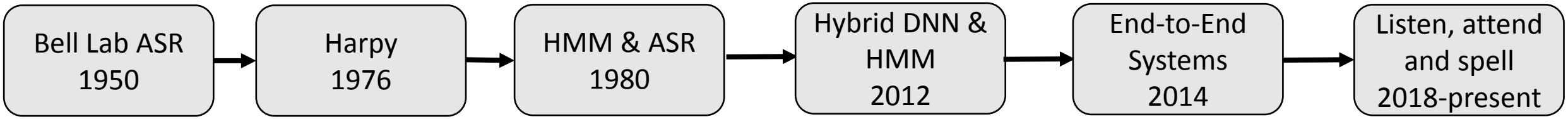
- 1952 – Automatic Digit Recognition (AUDREY)- Davis, Biddulph, Balashek (Bell Laboratories)
- Primul dispozitiv de recunoaștere a vorbirii serios a fost dezvoltat în 1952 la Bell Labs. S-au folosit circuite analogice folosind filtre, amplificatoare și integratori. Folosind un simplu divizor de frecvență, s-au generat diagrame ale primilor doi formanți, pe care i-a identificat prin potrivirea lor cu modele pre-stocate într-o memorie analogică. Cu antrenament, mașina a atins o precizie de 97% la cuvintele formate din cele zece cifre

Evoluția asistenților vocali



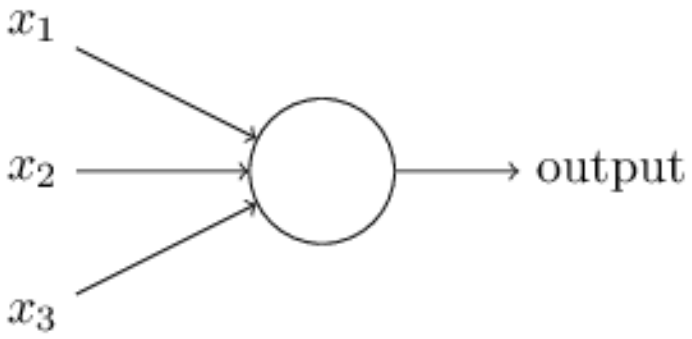
- Apple Siri (2011)
- Google Now (2012)
- Amazon Alexa (2014)
- Microsoft Cortana (2014)
- Până în prezent s-au realizat puține aplicații de recunoaștere, cele mai multe fiind dezvolate în laboratoarele de cercetare (UTCN, UPB, Academia Română)
- Cercetarea și studiul amănunțit sunt necesare în anumite domenii precum: domeniul medical, domeniul securității

State of the art



- Aplicații Speech-To-Text din piață
  - Google Cloud Speech-To-Text
  - Wav2vec-U (Facebook)
- DeepSpeech – sistem de recunoaștere dezvoltat de Mozilla
  - Arhitectură bazată pe rețele neuronale recurente (WER 11.85%)

Fundamentare Teoretică



Perceptronul

- Folosit pentru a rezolva probleme de clasificare
- Dezvoltat între anii 1950-1960 de către Frank Rosenblatt
- Primește ca intrări binare și produce o singură ieșire binară
- Pentru a determina ieșirea, Rosenblatt a introdus noțiunea de *ponderi* $(w_1, w_2, \dots)$
- Ieșirea neuronului este determinată de către următoarea relație:

$$output = \begin{cases} 0, \sum_j w_j x_j \leq prag \\ 1, \sum_j w_j x_j > prag \end{cases}$$

Neuronul Artificial (Neuronul Sigmoid)

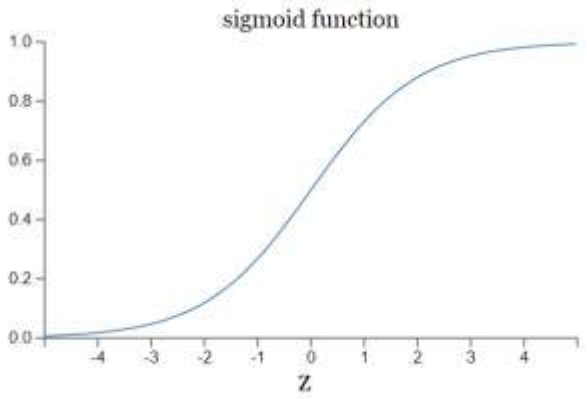
- Similar cu perceptronul
- Modificat -> schimbările mici din ponderi și prag cauzează doar o mică schimbare în ieșire -> permite învățarea cu o acuratețe mult mai bună decât prin utilizarea perceptronilor
- Diferența dintre perceptron și neuronul sigmoid este faptul că cel din urmă acceptă valori de intrare între [0,1]
- Ieșirea nu este 0/1, ci este: $\sigma(w * x + b)$, σ – funcția sigmoid

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

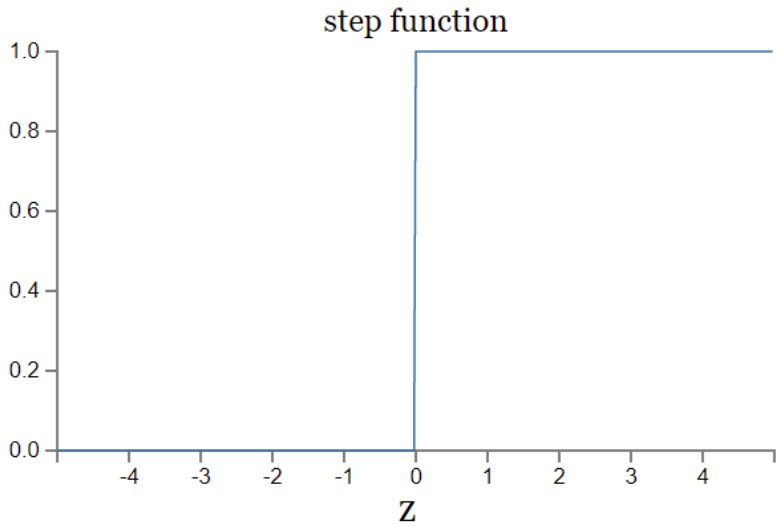
- Ieșirea unui neuron sigmoid care are ca intrare $x_1, x_2, \ldots$, ponderile $w_1, w_2, \ldots$, și bias-ul b ($b = -prag$):

$$\frac{1}{1 + \exp(-\sum_j w_j x_j - b)}$$

- Funcția de activare este responsabilă pentru transformarea ponderilor însumate de intrare a nodurilor în activarea nodului rezultat (de ieșire)



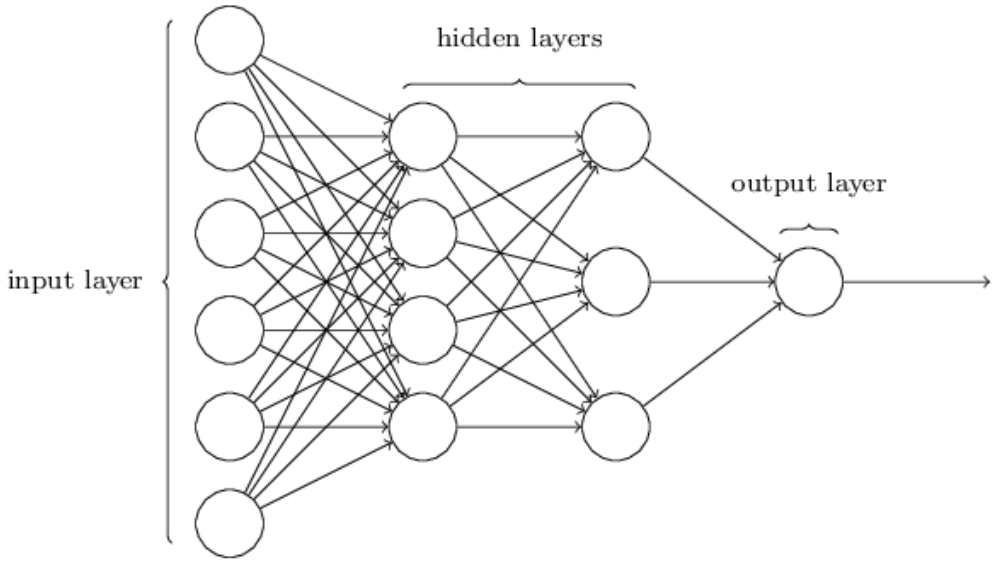
Funcția de activare a unui neuron sigmoid



Funcția de activare a unui perceptron

Rețele Neuronale (Multilayer Perceptrons)

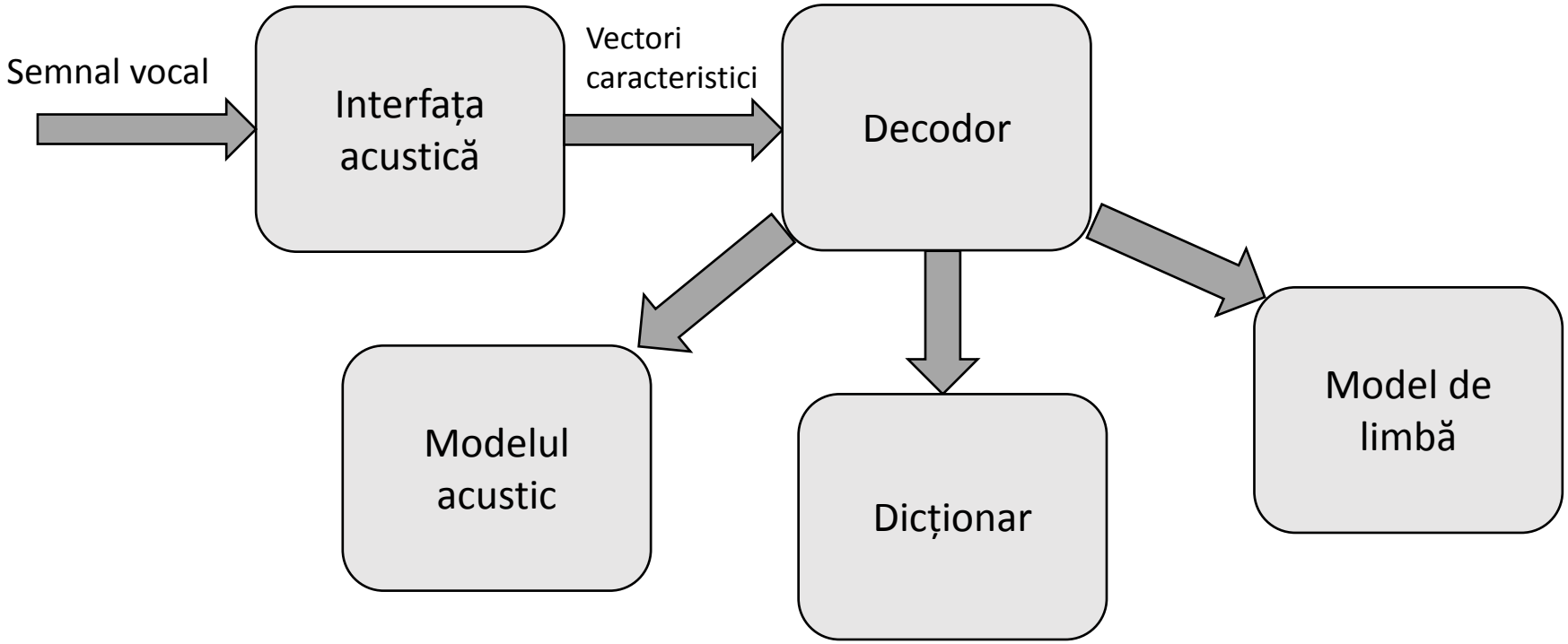
- Informația circulă de la stratul de intrare la straturile ascunse, iar apoi la stratul de ieșire
- Ieșirea unui strat este folosit ca intrare pentru stratul următor -> nu există bucle in rețea (informația circulă într-o singură direcție)



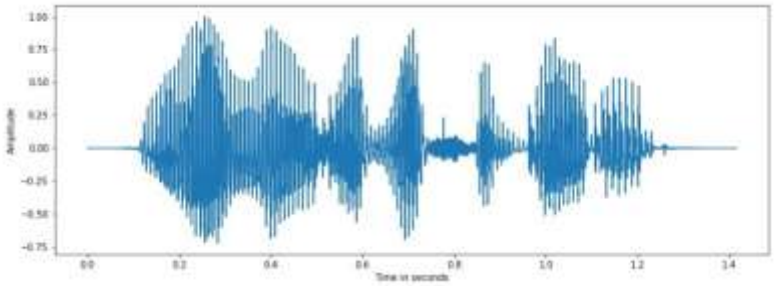
Rețele Recurente Neuronale

- Există și modele de rețele neuronale artificiale care permit bucle de informație (feedback loops) -> rețele recurente neuronale
- Prezintă “memorie” – preiau informația de la intrările precedente care afectează intrarea și ieșirea -> neuronii sunt activați în cascadă
- Buclele nu afectează acest model -> ieșirea unui neuron influențează intrarea după o anumită perioadă de timp, ci nu instantaneu!

Componentele unui sistem de recunoaștere a vorbirii



Interfața Acustică



Conversie Analog-digitală

Fereastră Hanning (20-30 ms)

- Conversie Scara Mel -> MFCC
- Semnalul este împărțit în ferestre (20-30 ms)
- Divizarea produce discontinuități => Fereastra Hanning
- Conversie domeniu timp -> frecvență

Fast-Fourier Transform (FFT)

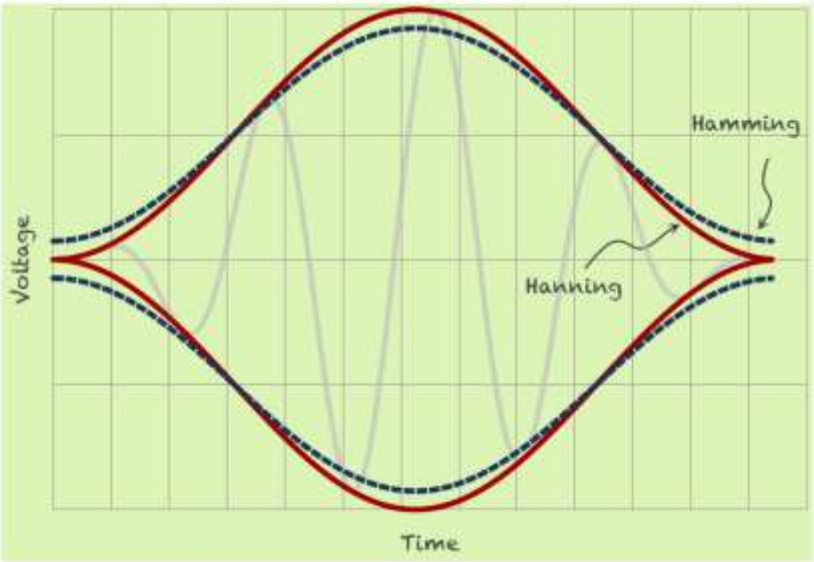
Mel Filter Bank (40 filtre triunghiulare)

Transformata Cosinus Discretă (DCT)

Vectori Caracteristici

Împărțirea în ferestre

- Metodă prin care semnalul este împărțit în segmente temporale (20-30 ms)
- Procesul de divizare produce discontinuități -> se realizează o multiplicare a semnalului de intrare cu o funcție fereastră (Hanning)



Transformata Fourier Rapidă

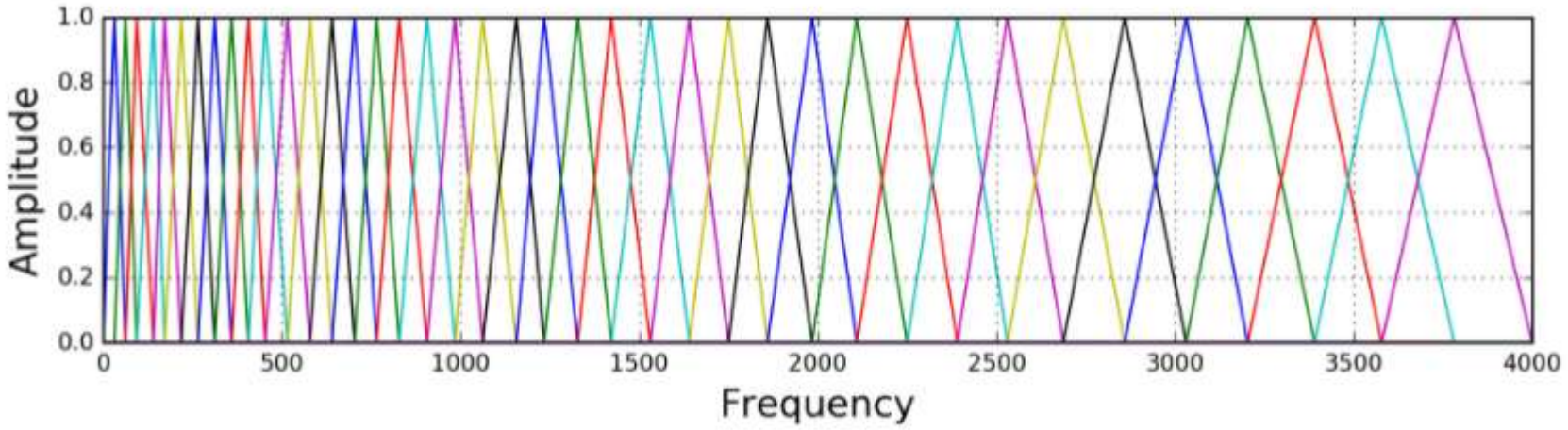
- Are rolul de a converti semnalul din domeniul timp în domeniul frecvență

$$X(k) = \sum_{j=1}^{N} x(j) e^{-\frac{2\pi i}{N}(j-1)(k-1)}$$

- $x(j)$ – reprezintă semnalul
- $k = 0,1, \dots, N-1, N$ – eșantioane la timpul j

Mel Filter Bank

- Rolul de a extrage benzi de frecvență
- Scara Male are rolul de a imita percepția umană a sunetului care este non-lineară: mult mai discriminativă la frecvențe mici, mai puțin discriminativă la frecvențe mari
- Se realizează o conversie de pe scala [Hz] pe scala [MEL]



Filter bank on a Mel-Scale

Transformata Cosinus Discretă Coeficienții MFCC

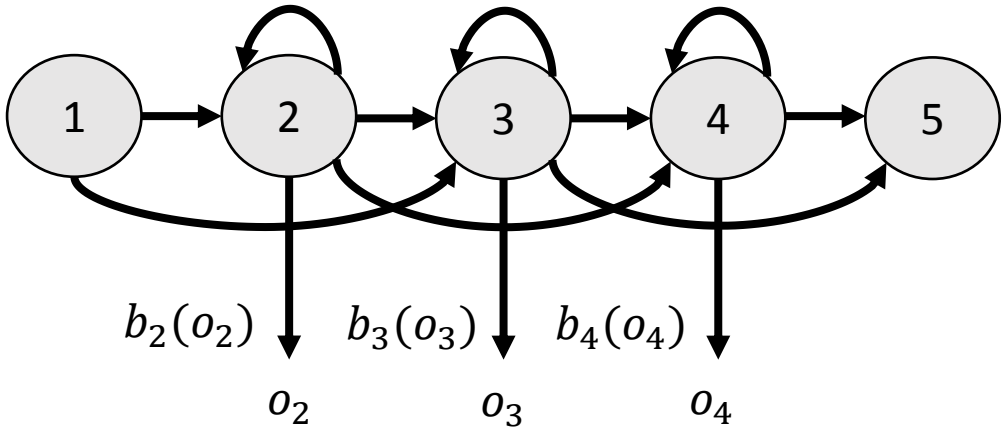
- Transformata Cosinus Discretă are rolul de a converti spectrumul obținut în domeniul timp -> Coeficienții spectrali pe scara Mel
- Datorită faptului ca informația semnalului este descrisă doar de primii coeficienți MFCC -> componentele DCT superior pot fi ignorate

$$c(n) = \sum_{m=0}^{M-1} \log_{10}\big(s(m)\big) \cos\left(\frac{\pi n(m-0.5)}{M}\right)$$

- $n = 0,1,2,\dots,C-1, c(n) - coeficienții\ spectrali$
- M – numărul de filtre triunghiulare

Modelul Acustic

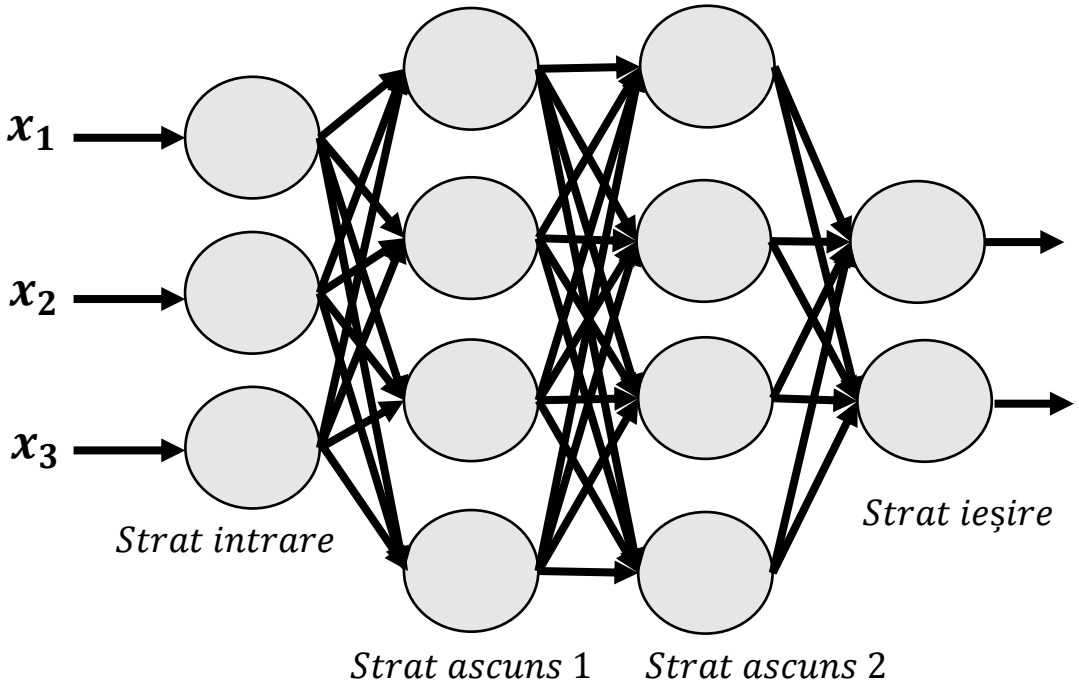
Model Markov Ascuns



(+) Fiabil de utilizat cand corpus-ul este foarte mic

(-) complicat de utilizat în recunoașterea automată a vorbirii
(-) necesită un timp mai îndelungat de învățare dacă stările ascunse au complexitate mare -> algoritm complex
(-) ridicarea complexității nu rezultă întotdeauna la o performanță mai bună a modelului

Rețea neuronală cu învățare profundă



(+) Performanțe bune când se utilizează un dataset mare
(+) Generalizare mai bună decât Modelul Markov Ascuns

Modelul de limbă

- Modelul de limbă este utilizat în
  - Recunoașterea vocală: P(mașina este roșie) > P(mașina este rodie)
  - Corecția de greșeli ortografice: P(arta înseamnă sacrificiu) > P(ata înseamnă sacrificiu)
- Modelul de limbă folosit de către DeepSpeech este modelul n-gram
- Modelul care determină probabilitatea cuvântului curent față de cuvintele precedente se numește modelul de limbă: $P(w_n|w_1, w_2, \dots, w_{n-1})$
- Modelul Unigram: $P(w_1 w_2 \dots w_n) \approx \prod_i P(w_i)$
- Modelul Bigram: $P(w_i|w_1 w_2 \dots w_{i-1}) \approx P(w_i|w_{i-1})$
- Cele două modele prezentate mai sus nu sunt insuficiente: propoziția se bazează pe context -> modelul N-gram

Modelul N-gram

- O metodă de a determina probabilitățile unei propoziții de forma $P(w_1, w_2, \dots, w_n)$ este prin folosirea regulii de înlănțuire a probabilităților

$$P(X_1 \dots X_n) = P(X_1)P(X_2|X_1)P(X_3|X_{1:2}) \dots P(X_n|X_{1:n-1}) = \prod_{k=1}^{n} P(X_k|X_{1:k-1})$$

- Ecuația precedentă evidențiază faptul că probabilitatea unei propoziții ar putea fi determinată prin multiplicarea unui număr de probabilități condiționale -> prin folosirea regulii de înlănțuire nu putem determina probabilitatea exactă a unui cuvânt dintr-o secvență lungă de cuvinte
- Intuiția modelului N-gram este că în loc de a determina probabilitatea unui cuvânt din toate cuvintele precedente, putem aproxima “istoria” utilizând doar câteva cuvinte precedente
- Modelul bi-gram aproximează probabilitatea unui cuvânt dat folosind doar probabilitatea cuvântului precedent $P(w_n|w_{n-1})$

- Normalizare prin unigrame: determinarea numărului de apariții în vocabular

Activitățile	de	aici	sunt	o	relaxare	pentru	noi
29	10623	309	1423	2750	14	1839	377

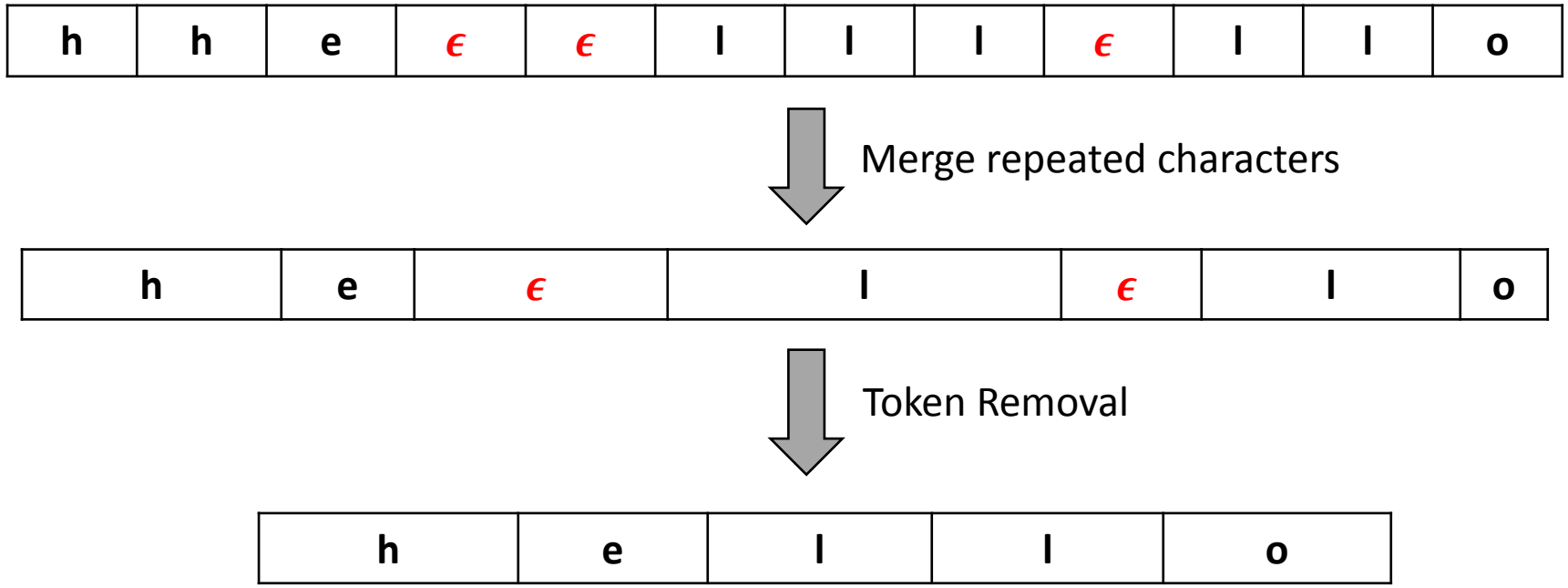
- Exemplu bigram folosind corpusul Swara (25337 propoziții)

	activitățile	de	aici	sunt	o	relaxare	pentru	noi
activitățile	0 0	28 0.96	0 0	0 0	0 0	0 0	0 0	0 0
de	0 0	1 0.009	78 0.0073	0 0	134 0.012	0 0	5 0.0047	4 0.003 7
aici	0 0	1 0.0032	0 0	31 0.100	2 0.0006	0 0	0 0	0 0
sunt	0 0	24 0.016	1 0.0070	0 0	25 0.0017	0 0	13 0.009	13 0.009
o	0 0	0 0	0 0	0 0	2 0.0072	14 0.0050	0 0	0 0
relaxare	0 0	0 0	0 0	0 0	0 0	0 0	14 1	0 0
pentru	0 0	0 0	0 0	0 0	29 0.0015	0 0	0 0	41 0.022
noi	0 0	5 0.013	1 0.0026	0 0	0 0	0 0	0 0	0 0

- $P(W_i|W_{i-1}) = \frac{count(W_{i-1},Wi)}{count(W_i)}$

Decodorul

- DeepSpeech utilizează decodorul CTC (Connectionist Temporal Classification)
- Are rolul de a alinia caracterele cu audio-ul corespondent

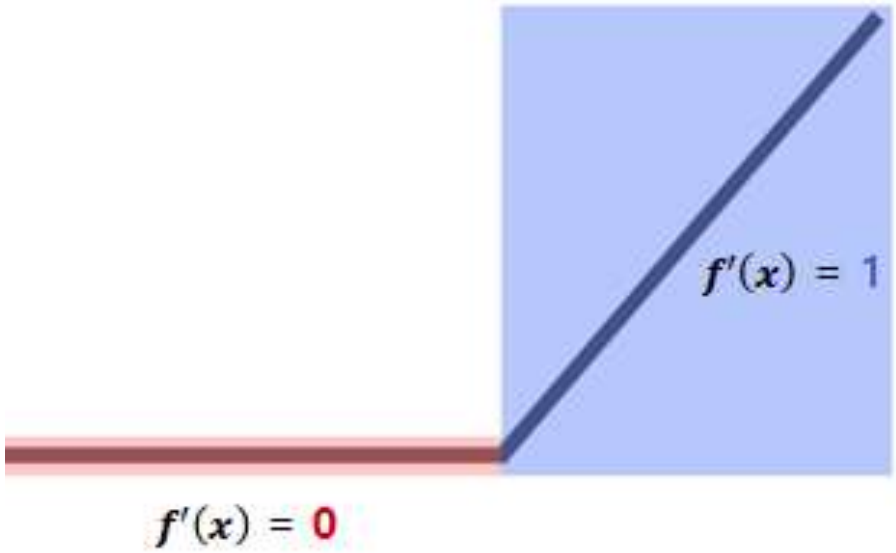


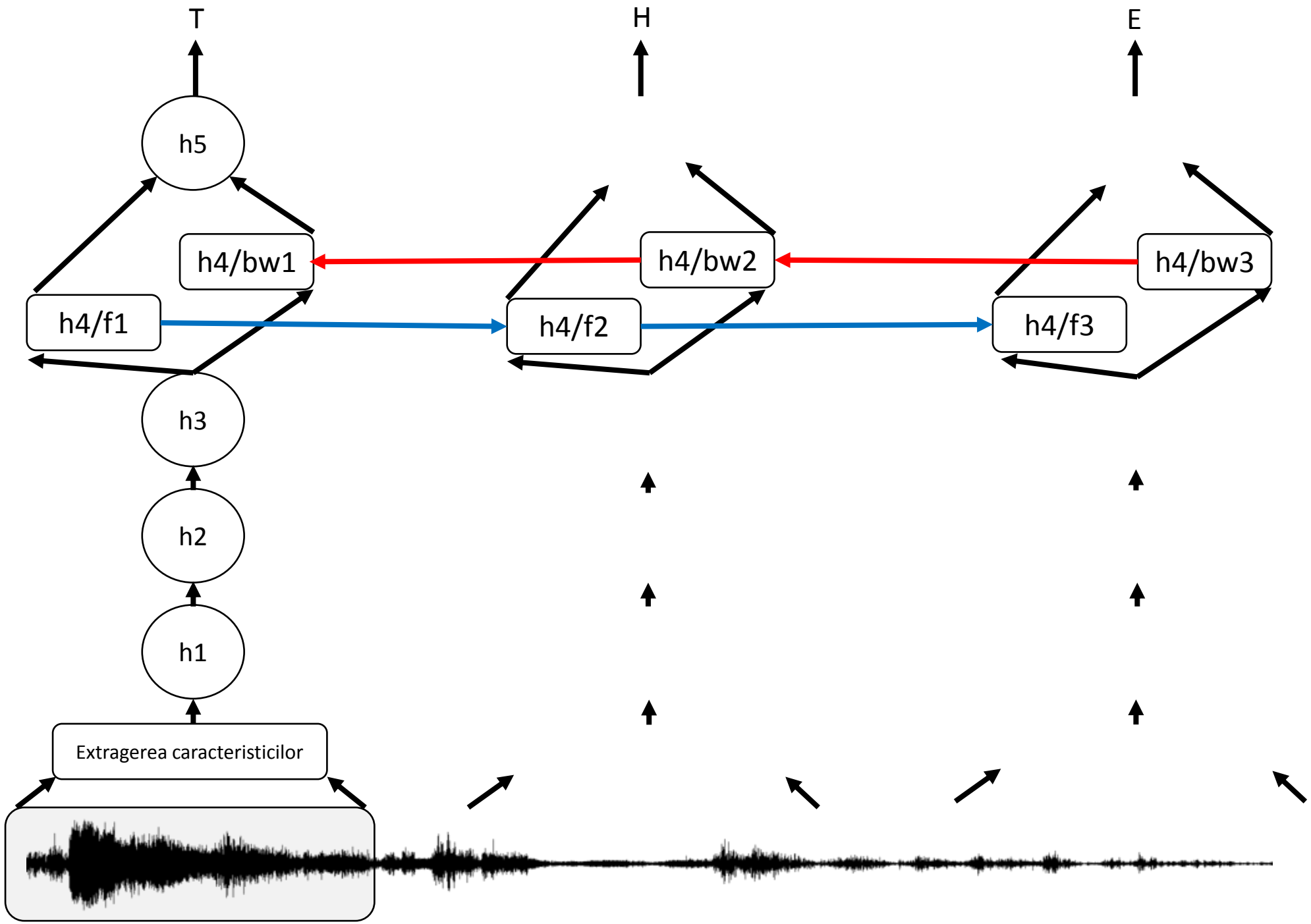
Implementarea Arhitecturii DeepSpeech

- Conține 5 straturi ascunse
- a) primele 3 straturi: Rețea full conectată cu funcția de activare ReLU

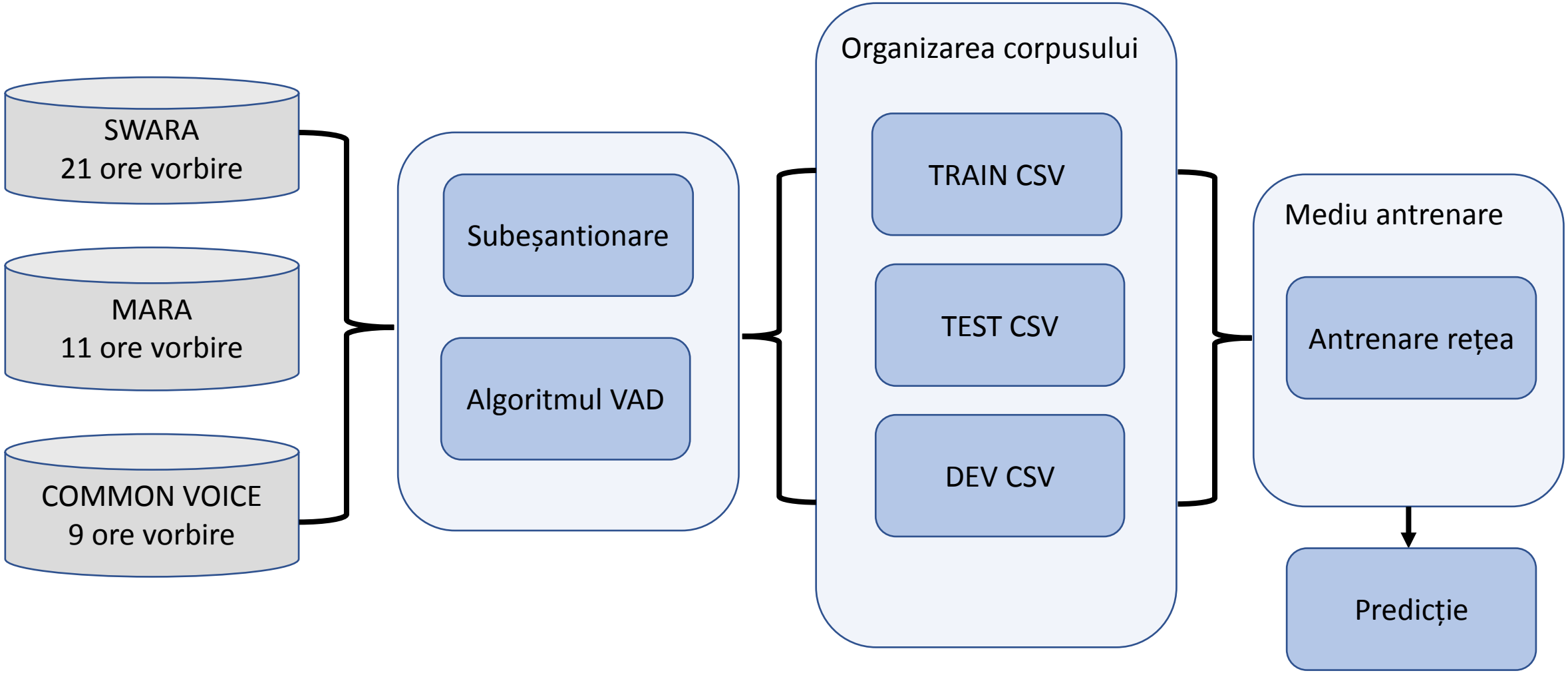
$$f(x) = \max(0, x)$$

- Funcția de activare ReLU (Unitatea liniară rectificată) -> ușor de antrenat și atinge performanțe mai bune
  - Straturile ascunse -> Antrenarea se realizează mai rapid
- b) al 4-lea strat: Rețea bidirecționala cu memorie
  - Filtrează și salvează informație pentru următoarele iterații
- c) al 5-lea strat: Rețea full conectată cu funcția de activare Softmax
  - Funcția de activare Softmax -> calculează distribuția probabilității la fiecare pas, input-ul este normalizat în intervalul (0,1) pentru procesul de decodare




T
H
E
h5
h4/bw1
h4/bw2
h4/bw3
h4/f1
h4/f2
h4/f3
h3
h2
h1
Extragerea caracteristicilor

Procesarea datelor



EXPERIMENT 1
Corpus SWARA fără model de limbă

$$\text{Word Error Rate} = \frac{\text{Insertions} + \text{Deletions} + \text{Substitutions}}{\text{Number of Words in Reference Transcript}}$$

- Rata totală de eroare a cuvintelor (WER) = 46.4 %
- Rata totală de eroare de caractere (CER) = 13.48%
- Pierderea totală (Loss) = 71.12%
- CER – metrică pentru determinarea erorilor la nivel de caracter
- WER – metrică pentru determinarea erorilor la nivel de cuvânt

$$CER = \frac{S + D + I}{N}$$

Best WER	Medium WER	Worst WER
Src=” muniția era de provenienţă germană” Res=”muniția era de provenienţă germană”	Src=”eu lucrez din o mie nouă sute nouăzeci și doi în domeniu dar nu am mai întâlnit așa ceva” Res=”eu lucrez din uie nu spe nozdeci și doi în domeniu dar nu m mintunite șa ceva”	Src=” curtea lăcașului sfânt a devenit neîncăpătoare” Res=” cute la gașuli s sunt a devet de încăpătare”
WER: 0.0	WER:0.47	WER: 1.33
CER: 0.0	CER: 0.20	CER: 0.30
Loss: 1.54	Loss: 203.25	Loss: 112.07

EXPERIMENT 2
Corpusul SWARA cu model de limbă și validare

- Rata totală de eroare a cuvintelor (WER) = 47.16 %
- Rata totală de eroare de caractere (CER) = 21.22%
- Pierderea totală (Loss) = 53.76%

Best WER	Medium WER	Worst WER
Src=” anul acesta de exemplu a fost un an foarte bun” Res=” anul acesta de exemplu a fost un an foarte bun”	Src=” a fost foarte greu pentru ea în ultima vreme” Res=” a fost foarte greu pentru de animalele”	Src=” arta înseamnă sacrificiu” Res=” dat seama sa criticul”
WER: 0.0	WER:0.44	WER: 1.4
CER: 0.0	CER: 0.20	CER: 0.41
Loss: 39.77	Loss: 33.71	Loss: 11.87

EXPERIMENT 3
Corpusul SWARA cu model de limbă bazat pe subtitrări de filme

- Rata totală de eroare a cuvintelor (WER) = 19.65 %
- Rata totală de eroare de caractere (CER) = 7.3%
- Pierderea totală (Loss) = 44.74%

Best WER	Medium WER	Worst WER
Src=” anul acesta de exemplu a fost un an foarte bun” Res=” acesta de exemplu a fost un an foarte bun”	Src=” a fost foarte greu pentru ea în ultima vreme” Res=” a fost foarte greu pentru ei în ultima vreme”	Src=” arta înseamnă sacrificiu” Res=” arta înseamnă sacrificiu”
WER: 0.1	WER:0.11	WER: 0.0
CER: 0.10	CER: 0.02	CER: 0.0
Loss: 19.6	Loss: 6.51	Loss: 0.83

EXPERIMENT 4
SWARA+MARA+COMMON VOICE+model de limbă bazat pe corpusul vocabularului

- Rata totală de eroare a cuvintelor (WER) = 23.81 %
- Rata totală de eroare de caractere (CER) = 10.17%
- Pierderea totală (Loss) = 80.30%

Best WER	Medium WER	Worst WER
Src=” anul acesta de exemplu a fost un an foarte bun” Res=” acesta de exemplu a fost un an foarte bun”	Src=” a fost foarte greu pentru ea în ultima vreme” Res=” a fost foarte greu pentru e în ultima vreme”	Src=” arta înseamnă sacrificiu” Res=” arta înseamnă sa critici”
WER: 0.1	WER:0.11	WER: 0.6
CER: 0.10	CER: 0.02	CER: 0.125
Loss: 133.56	Loss: 4.65	Loss: 16.56

EXPERIMENT 5
SWARA+MARA+COMMON VOICE+model de limbă bazat pe subtitrări de filme

- Rata totală de eroare a cuvintelor (WER) = 29.93 %
- Rata totală de eroare de caractere (CER) = 11.67%
- Pierderea totală (Loss) = 84.53%

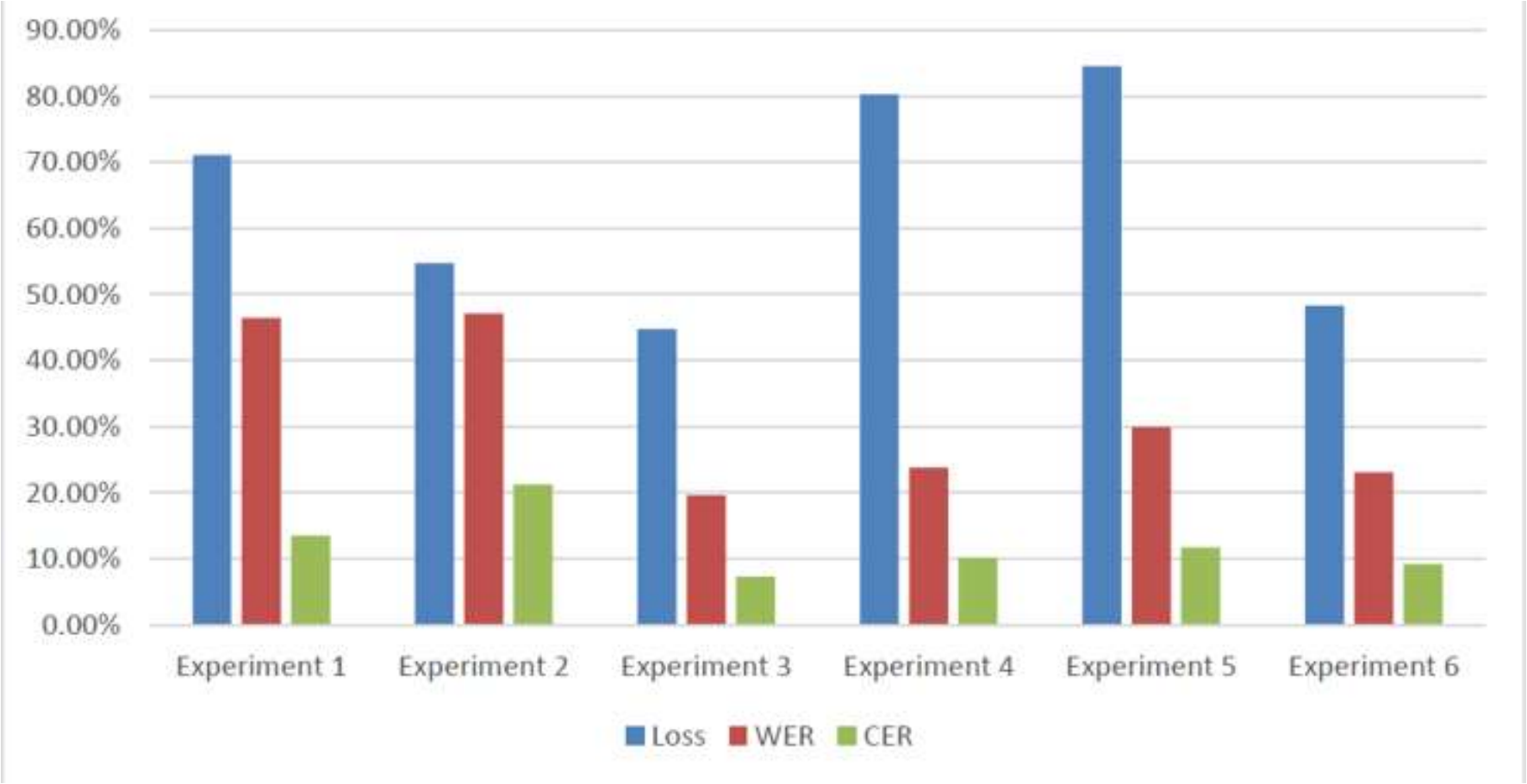
Best WER	Medium WER	Worst WER
Src=” anul acesta de exemplu a fost un an foarte bun” Res=” acesta de exemplu a fost un an foarte bun”	Src=” a fost foarte greu pentru ea în ultima vreme” Res=” a fost foarte greu pentru e în ultima vreme”	Src=” arta înseamnă sacrificiu” Res=” arta nseamnă sacrificiu”
WER: 0.1	WER:0.11	WER: 0.33
CER: 0.10	CER: 0.02	CER: 0.04
Loss: 133.56	Loss: 4.65	Loss: 16.56

EXPERIMENT 6
SWARA+MARA+COMMON VOICE+AUGUMENTARE+model de limbă bazat pe subtitrări de filme

- Augumentare
- Rata totală de eroare a cuvintelor (WER) = 29.93 %
- Rata totală de eroare de caractere (CER) = 11.67%
- Pierderea totală (Loss) = 84.53%

Best WER	Medium WER	Worst WER
Src=” anul acesta de exemplu a fost un an foarte bun” Res=” acesta de exemplu a fost un an foarte bun”	Src=” a fost foarte greu pentru ea în ultima vreme” Res=” a fost foarte greu pentru e în ultima vreme”	Src=” arta înseamnă sacrificiu” Res=” arta nseamnă sacrificiu”
WER: 0.1	WER:0.11	WER: 0.33
CER: 0.10	CER: 0.02	CER: 0.04
Loss: 133.56	Loss: 4.65	Loss: 16.56

Sinteza celor 6 experimente



Concluzii

- Dezvoltarea unui sistem de recunoaștere a vorbirii utilizând DeepSpeech
- Înțelegerea și implementarea arhitecturii DeepSpeech
- Implementarea diferitor experimente pentru îmbunătățirea performanței modelului
- Dezvoltare viitoare:
  - Îmbunătățirea corpusului cu mai multe date
  - Adăugare de corpus cu zgomot pentru o mai bună generalizare
  - Mai multă augumentare

Bibliografie

- https://web.stanford.edu/~jurafsky/slp3/
- http://neuralnetworksanddeeplearning.com/chap1.html
- https://deepspeech.readthedocs.io/en/r0.9/
- https://github.com/jacobeisenstein/gt-nlp-class/blob/master/notes/eisenstein-nlp-notes.pdf

Click to add text