Robot hangvezérlő rendszer alapjául szolgáló neurális hálózat tervezése
Design of neural network as a basis of a robot voice-control system

Kiss Gábor

Kivonat

Az automatikus hangfelismerés terén komoly előre lépések történtek az elmúlt években, részben a neurális hálózatoknak köszönhetően. Sok megoldás született szavak felismerésére angol és kínai nyelveken, azonban magyar nyelven ez nem jellemző. Feladatom során több hangfelismerő modellt készítettem konvolúciós, rekurrens és előrecsatolt neurális hálózatok felhasználásával, amelyek képesek a meghatározott szavak felismerésére, beszélőtől függetlenül úgy, hogy viszonylag kis méretű adathalmazon történik a tanításuk. A hálózatok között szerepelnek olyan megoldások is, amik nem igényelnek előzetes feldolgozást, úgynevezett end-to-end rendszerként működnek. Dolgozatom célja, hogy összehasonlítsam ezeket a hálózatokat több szempont szerint is, többek között a pontosság és a hálózat paramétereinek száma alapján. Mivel a végső célom robotok hangvezérlésére alkalmas modell készítése, fontos szempontként kezelem a hálózat méretét, ugyanis túl nagy méretű hálózat futtatása túlságosan számítási kapacitás igényes lehet. Az összehasonlításból kiderül, hogy az MFCC algoritmus segítségével előkészített adatokat használó hálózatok általánosságban jobban teljesítenek, state-of-the-art pontosságot hoznak.

Abstract

In the recent years there were significant improvements in the area of automatic speech recognition, partly because of the neural networks. There are plently of new solutions to recognise words in English and Chniese, but it is not the case in hungarian. I created multiple ASR modell based on convolutional, recurrent and feedforward neural networks, which are albe to recognise word regardless of the speaker. Amongs my solutions there are some, which do not need any data preprocessing, work as end-to-end systems. The aim of my work to compare the networks according to multiple indicators, e.g. accuracy, number of parameters. The final goal is to create a robot voice-control system, the size of the neural network is a really important parameter here, because appliaction of networks with a lots of parameters can be too computation-expensive. The comparison reveals that the networks using MFCC pre-processed data bring better performance.

***Kulcsszavak*— Automatikus hangfelismerés, Konvolúciós neurális hálózatok, LSTM, Neurális hálózatok**

I. Bevezetés

A neurális hálózatok térhódítása a közelmúltban jelentősen kiterjedt a hangfelismerés területére is. Ez a terület egy meglehetősen bonyolult és összetett ága, ugyanis a hangok nagyon széles frekvencia és amplitúdó tartományban mozoghatnak. Emellett jelentősen befolyásolja a felvételek minőségét, hogy milyen eszközzel vesszük fel őket, illetve mennyire zajos a környezet, a jel mekkora része értékes adat. Mindezeken túl nagy mennyiségű adatról is beszélünk, ami tovább nehezíti a feladatot.

Ezeknek a problémáknak a jelentős részét célozza meg az adatelőkészítés, ami a state-of-the-art hangfelismerő modelleknek is szerves része alapján. A leggyakrabban és legeredményesebben alkalmazott módszer [1] [2] [3] [4] [5] alapján az MFCC transzformáció, melynek során a hangot reprezentáló időfüggvényt ablakokra bontjuk és az egyes ablakoknak nézzük bizonyos frekvencia tartománybeli jellemzőjét. Az így előkészített adatokat konvolúciós hálózat segítségével remekül lehet további absztrakciós szintekre emelni, ugyanis ezek a hálózatok lokálisan keresnek összefüggéseket az adatsoron, szemben az előrecsatolt hálózatokkal.

Kezdenek azonban teret hódítani az end-to-end rendszerek is, amiknek a célja a teljes felismerési folyamat megvalósítása. [6] [7] alapján azonban ezek a rendszerek is előkészített adatokkal dolgoznak legtöbb esetben. Dolgozatomban megvizsgálom, hogy a

minimális előkészítéssel kezelt adatok alapján milyen WER (Word Error Rate, elhibázott szavak aránya) képesek hozni az LSTM hálózatok és a [8] [9] alapján felállított konvolúciós, rekurrens és előrecsatolt rétegeket is tartalmazó hálózat.

II. Adatgyűjtés, adatelőkészítés

A. *Adatgyűjtés*

A hálózatok tanításához szükséges adatok beszerzése nem egyszerű feladat, ugyanis nem találtam elérhető, magyar nyelvű adatbázist, ami megfelelő adatokat tartalmazott volna, így az adatgyűjtést magam végeztem. Az adatgyűjtés során 1 másodperc hosszú, 44.1 kHz mintavételezésű hangfájlokat készítettem megfelelő névvel ellátva, ezzel is könnyítve a későbbi feldolgozásukat. Mind a 15 választott szót (menj, gyere, tölts, Ethon, vezess, keress, szia, vissza, nyisd, viszlát, Bea, Ádám, Márta, Antal, Bence) felolvastattam 15 női és 11 férfi önkéntessel. Minden beszélő minden szót ötször olvasott fel, ezzel szavanként 130 mintát, összesen 1950 felvételt eredményezve. A felvételek egy egyszerű laptop mikrofonnal készítettem egy kollégiumban, így a modelleknek a zajhatásokat is figyelembe kell venniük, ki kell szűrniük.

B. *Adatelőkészítés*

Az adatsorokat beszélők szerint elkülönítettem és 19 beszélő adatai alapján tanítottam, 4 beszélő, azaz 300 adatsor alapján validáltam és 3 beszélő felvételein teszteltem a hálózatokat. Összehasonlításként kipróbáltam egy olyan felosztást is, ahol minden hangfájlt beolvastam, összekevertem és utána osztottam fel tanító, validációs és teszt adathalmazokra is, hogy lássam, mennyire tanulja meg a hálózat az egyes beszélők hangját, beszédstílusát.

Ahogy már említettem, alapvetően kettő féle adatelőkészítést alkalmaztam. Először a gyakorlatban bevett MFCC transzformáció segítségével, ahol 25 ms széles ablakokat vettem 10 ms léptetéssel és Hamming ablakolással [3] [9] alapján. Frekvencia tartományban 20 szűrőt alkalmaztam, valamint első és másodrendű deriváltakat is. Ez egy 2D tömb formátumot eredményez, ami egyaránt használható 2D és 1D konvolúció, valamint LSTM hálózatok bemeneteként is. Ezt követően az egyes bemenő paramétereket nulla várható értékű, egységnyi szórású adatsorokká alakítottam úgy, hogy az ehhez szükséges paramétereket kizárólag a tanító adathalmaz alapján számítottam.

Az end-to-end modellezésre merőben más adatelőkészítést használtam. Ehhez mindössze felbontottam a jelet 20 ms hosszú ablakokra 10 ms lépésközzel. Ez szintén egy 2D tömböt eredményez, ami használható konvolúciós és LSTM hálózatok bemeneteként is. Az adatsoron nem végeztem semmilyen szűrést ebben az esetben, a zajos jelben a lényegkiemelés is a hálózatok feladata.

III. Betanított modellek felépítése

Ahogyan azt már említettem is, több különböző struktúrájú hálózatot hoztam létre, hogy összehasonlíthassam őket. Az első modell, amit felépítettem [4] [9] alapján, MFCC-vel előkészített adatokat használ. Felépítését tekintve többrétegű, 2D konvolúciós rétegekből és előrecsatolt rétegekből, valamint a konvolúciós rétegek közt [8] [4] [5] alapján max pooling és BatchNormalization rétegek is vannak. Ezt a hálózatot a továbbiakban CNN+Dense_sep néven hivatkozom.

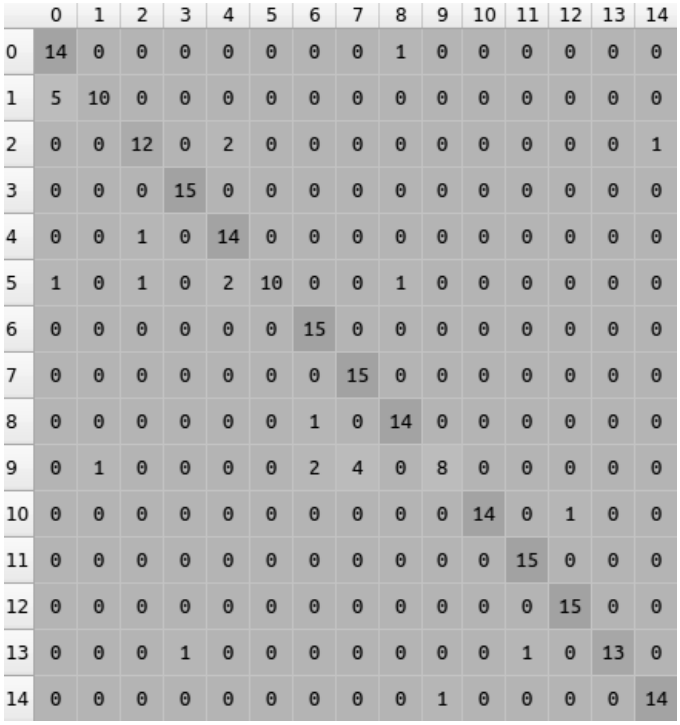
Ennek a hálózatnak egy kis mértékben módosított változatát hoztam létre arra a célra, hogy a nem beszélőnként szeparált adatsorokat vizsgáljam. Ez a hálózat is konvolúciós rétegekből, pooling rétegekből, BatchNormalization rétegekből és előrecsatolt rétegekből áll, azonban egyel kevesebb előrecsatolt réteget tartalmaz. Erre a hálózatra a továbbiakban CNN+Dense néven hivatkozom.

Az end-to-end felhasználásra először többrétegű LSTM és előrecsatolt rétegekből álló hálózatot hoztam létre, mivel [7] alapján az LSTM hálózatok alkalmasak időbeli összefüggések felismerésére. Ezt a hálózatot LSTM+Dense néven hivatkozom a továbbiakban.

Annak vizsgálatára, hogy egy bizonyos struktúrájú hálózat az előkészített vagy a nyers adatokat képes jobban megtanulni és értelmezni, létrehoztam egy 1D konvolúciós rétegekből, LSTM rétegekből és előrecsatolt rétegekből álló hálózatot is a konvolúciós rétegek között pooling és BatchNormalization rétegekkel. Természetesen a kettő hálózat különbözik valamelyest a paraméterek számában, így az előkészített adatokkal dolgozó hálózatot CNN+RNN+Dense_sep_MFCC néven, míg az end-to-end hálózatot CNN+RNN+Dense_sep néven hivatkozom a továbbiakban.

A hálózatok létrehozását és tanítását Keras keretrendszer segítségével végeztem és minden esetben szekvenciális modellt használtam. A tanítás során validációs adathalmazt és early stopping eljárást használtam 30 epoch türelmi idővel.

Mivel osztályozási feladatról van szó, az utolsó réteg aktivációja minden esetben Softmax függvény volt. Költségfüggvényként kereszt-entrópia függvényt választottam és ADAM optimalizációs algoritmust a backpropagation tanításhoz.



1. ábra CNN+Dense_sep tévesztési mátrix

IV. KIÉRTÉKELÉS SZEMPONTJAI

A kiértékelés során több szempontot is figyelembe veszek, azonban mindig csak a teszt adathalmazon nyújtott teljesítményt vizsgálom. Az első szempont a hálózat hibája, költsége a költségfüggvény alapján. Mivel a tanítás is ez alapján történik, jó mérőszám a hálózat jellemzésére, azonban önmagában nem elegendő.

A költség mellett vizsgálom a hálózat pontosságát, vagyis a helyes predikciók arányát az összes predikcióhoz képest. Ez az egyik legfontosabb jellemzője a hálózatnak, hiszen a célja a megfelelő osztályba sorolás.

Ezzel párhuzamosan konfúziós, vagy tévesztési mátrixot is készítek minden hálózatról, ami a legbeszédesebb indikátor abban a viszonylatban, hogy az egyes szavakat milyen gyakran keveri össze a hálózat. Ez alapján könnyen detektálhatók azok a szópárok, amik használata lehetőség szerint kerülendő.

Ezek mellett vizsgálom minden szóra külön a precizitást, a felidézést és az f1 mérőszámot is. A precizitás a helyes predikciók és az összes predikció hányadosa az adott szóra. A felidézés mérőszáma mutatja meg a helyes predikciók és a maximálisan helyes predikciók számát az adott szóra, az f1 mérőszám pedig precizitás és a felidézés harmonikus közepe.

Emellett, ahogy már említettem fontos indikátorként tartom számon a hálózat paramétereinek a számát is, ugyanis végső célom egy robot hangvezérlő rendszer fejlesztése.

1. TÁBLÁZAT
AZ EGYES HÁLÓZATOK KERESZT-ENTRÓPIA KÖLTSÉGE

Hálózat neve	Kereszt-entrópia költség
CNN+Dense_sep	0.402
CNN+Dense	0.689
LSTM+Dense	2.299
CNN+RNN+Dense_sep_MFCC	0.823
CNN+RNN+Dense_sep	0.885

2. TÁBLÁZAT
AZ EGYES HÁLÓZATOK ÁTLAGOS PRECIZITÁSA, FELIDÉZÉSE ÉS F1 MÉRŐSZÁMA

Hálózat neve	Precizitás	Felidézés	f1
CNN+Dense_sep	0.89	0.88	0.88
CNN+Dense	0.88	0.86	0.86
LSTM+Dense	0.30	0.28	0.27
CNN+RNN+Dense_sep_MFCC	0.86	0.83	0.83
CNN+RNN+Dense_sep	0.75	0.72	0.72

3. TÁBLÁZAT
AZ EGYES HÁLÓZATOK PARAMÉTEREINEK SZÁMA

Hálózat neve	Paraméterek száma
CNN+Dense_sep	295,725
CNN+Dense	214,601
LSTM+Dense	3,173,215
CNN+RNN+Dense_sep_MFCC	6,085,515
CNN+RNN+Dense_sep	6,193,265

4. TÁBLÁZAT
AZ EGYES HÁLÓZATOK PONTOSSÁGA

Hálózat neve	Pontosság [%]
CNN+Dense_sep	88.0
CNN+Dense	86.15
LSTM+Dense	27.55
CNN+RNN+Dense_sep_MFCC	82.66
CNN+RNN+Dense_sep	72.44

V. TESZT EREDMÉNYEK

A. Kereszt-entrópia költség

Az első szempont, ami alapján a hálózatokat összehasonlítom a kereszt-entrópia függvénnyel számolt költség a teszt adathalmazon. Értelem szerűen ez az érték minél kisebb, annál nagyobb biztonsággal prediktál megfelelő eredményt a hálózat. Ezeket az értékeket 1. Táblázat foglalja össze. Jól látható, hogy a kettő legjobban teljesítő hálózat 2D konvolúciót használ és MFCC előkészítést. Ezek a hálózatok közül is az

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
0	15	1	1	0	0	0	0	0	0	0	0	0	0	0	0
1	0	14	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	12	0	1	0	0	0	0	0	0	0	0	0	0
3	0	0	0	14	0	0	0	0	0	0	1	0	0	0	0
4	0	0	4	0	4	2	0	0	0	0	0	0	0	0	0
5	0	1	1	0	0	13	0	0	0	0	0	0	0	0	1
6	0	0	0	0	0	0	11	1	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
8	0	0	0	0	0	0	1	0	14	0	0	0	0	0	0
9	0	1	0	0	0	0	1	3	0	7	0	0	0	0	1
10	0	0	0	1	0	0	0	0	0	0	11	0	0	0	0
11	0	0	0	0	1	0	0	0	0	0	0	15	1	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	9	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	2	11	0
14	0	1	0	0	0	0	0	0	0	0	0	0	0	0	10

2. ábra CNN+Dense tévesztési mátrix

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
0	4	4	0	3	0	0	0	0	2	0	1	0	1	0	0
1	5	4	1	1	0	0	1	0	1	1	0	1	0	0	0
2	1	3	4	0	0	1	1	0	3	0	0	0	1	0	1
3	3	1	0	3	0	0	0	0	0	2	5	1	0	0	0
4	0	0	0	0	5	4	0	0	4	0	0	0	1	0	1
5	0	1	1	0	2	5	0	0	3	1	0	0	0	0	2
6	2	1	0	0	0	0	0	1	1	8	1	0	1	0	0
7	0	0	0	0	1	0	0	5	1	5	0	0	0	0	3
8	2	4	0	0	1	0	0	1	4	2	0	0	0	0	1
9	0	1	0	1	0	0	1	3	0	6	0	0	0	0	3
10	1	1	0	3	0	0	1	0	0	0	4	2	2	1	0
11	1	2	0	2	0	0	0	0	0	1	0	4	4	1	0
12	0	1	0	1	0	0	0	1	0	0	2	0	7	3	0
13	0	1	1	3	0	0	0	0	0	1	2	2	3	2	0
14	2	0	1	1	0	0	1	3	1	1	0	0	0	0	5

3. ábra LSTM+Dense tévesztési mátrix

teljesített jobban, amelyiknek az adatsorát beszélőnként szeparáltam. Ez jól mutatja, hogy a használt struktúra nem a beszélők hangját és beszédstílusát, hanem az egyes szavakhoz tartozó hangtani jelenségeket tanulja meg, ahogyan azt el is várnánk.

A legrosszabbul az LSTM alapú hálózat teljesített. Ennek nagy valószínűséggel az az oka, hogy struktúrájából adódóan lényegesen több tanító mintára lenne szüksége a hálózatnak, kiváltképpen az előkészítés elhagyása miatt.

Az előkészítés fontosságának vizsgálatára létrehozott struktúrák között jobban teljesített az, amelyik MFCC adatokkal dolgozott, azonban nem sokkal marad el tőle az előkészítetlen adatokkal dolgozó hálózat sem a költség tekintetében. Csak ez a mérőszám azonban nem alkalmas a hálózatok teljes összehasonlítására, végleges konklúziót ebből még nem szabad levonni.

B. Precizitás, felidézés, f1

Ezen mérőszámokkal jellemzett eredményeket 2. Táblázat tartalmazza. Jól látható, hogy ebben a tekintetben is a 2D konvolúció alapú hálózatok teljesítettek a legjobban, azonban jelentősen megközelítette őket a *CNN+RNN+Dense_sep_MFCC* hálózat is. A precizitásban mutatkozó minimális különbség, ami az egyik legfontosabb mérőszáma egy osztályozó hálózatnak, megmutatja, hogy a feladat megoldására több, különböző struktúra is hasonlóan jó megoldást jelenthet. Kérdéses, hogy ezek futási időben és robosztusságban hogyan viszonyulnak egymáshoz. Ennek vizsgálatára több, változatos környezetből felvett teszt mintára lenne szükség.

CNN+RNN+Dense_sep_MFCC hálózat és a *CNN+RNN+Dense_sep* hálózat közül most is az előkészített adatokkal dolgozó jött ki győztesen. Mind a három mérőszám tekintetében több, mint 10%-kal teljesített jobban. Ez egy igen jelentős különbség és remekül mutatja, hogy nem szabad csak a költségfüggvényre támaszkodni egy hálózat megítélésekor.

A legrosszabbul ezek a szempontok szerint is az LSTM+Dense hálózatt végzett 30%-os átlagos precizitással. Ez az érték nagyon alacsony, de mégis azt mutatja, hogy valamilyen szinten képes volt a hálózat hangtani összefüggéseket megtanulni. Valószínűleg nagyobb tanító adathalmazzal sikeress lenne a tanítás.

C. Paraméterek száma

A 3. Táblázat mutatja, hogy melyik hálózat mennyi paraméterrel rendelkezik. Jól látható, hogy a 2D konvolúció alapú hálózatok sokkal kevesebb paramétert használnak, mint azok, amik LSTM rétegeket is tartalmaznak. Ez a konvolúciós hálózatok felépítéséből adódik.

A többi hálózat egy nagyságrenddel több paraméterrel dolgozik, ami erősen korlátozza felhasználhatóságukat korlátozott számítási kapacitású eszközökön. Ebben a tekintetben egyértelműen egyeduralkodóak a konvolúció alapú hálózatok.

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
0	12	0	1	0	0	0	0	0	2	0	0	0	0	0	0
1	2	8	0	0	0	0	0	0	5	0	0	0	0	0	0
2	0	0	15	0	0	0	0	0	0	0	0	0	0	0	0
3	2	0	0	10	0	0	0	0	0	0	1	0	1	1	0
4	0	0	3	0	12	0	0	0	0	0	0	0	0	0	0
5	0	1	0	0	0	9	0	0	5	0	0	0	0	0	0
6	0	0	0	0	0	0	12	1	0	2	0	0	0	0	0
7	0	0	0	0	0	0	0	15	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	15	0	0	0	0	0	0
9	0	0	0	0	0	0	0	4	0	11	0	0	0	0	0
10	0	0	0	0	0	0	0	1	0	0	11	2	1	0	0
11	0	0	0	0	0	0	0	0	0	0	0	15	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	15	0	0
13	0	0	0	1	0	0	0	0	0	0	0	0	2	12	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	1	14

4. ábra CNN+RNN+Dense_sep_MFCC tévesztési mátrix

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
0	12	1	0	0	0	0	0	0	2	0	0	0	0	0	0
1	3	4	1	0	0	0	2	0	0	0	5	0	0	0	0
2	1	0	11	0	0	3	0	0	0	0	0	0	0	0	0
3	0	0	0	10	0	0	0	0	1	0	0	1	0	3	0
4	0	0	0	0	12	3	0	0	0	0	0	0	0	0	0
5	1	0	0	0	3	8	0	0	3	0	0	0	0	0	0
6	0	2	0	0	0	0	12	0	1	0	0	0	0	0	0
7	0	0	0	0	0	0	1	14	0	0	0	0	0	0	0
8	0	0	0	0	0	0	1	0	14	0	0	0	0	0	0
9	0	0	0	0	2	0	4	2	0	7	0	0	0	0	0
10	3	0	0	0	0	0	0	0	0	0	12	0	0	0	0
11	1	0	0	0	0	0	0	0	0	0	0	10	3	1	0
12	0	0	1	0	0	0	0	0	0	0	0	3	10	1	0
13	1	0	0	0	0	0	0	0	0	0	0	0	0	14	0
14	0	0	1	0	0	0	0	0	1	0	0	0	0	0	13

5. ábra CNN+RNN+Dense_sep tévesztési mátrix

D. Pontosság

Ez az egyik elgfontosabb mérőszáma egy osztályozó hálózatnak, hiszen megadja a helyes predikciók arányát. A teszt eredményeit a 4. Táblázat tartalmazza.

Ebben a tekintetben is kiemelkednek a 2D konvolúción alapuló hálózatok, azonban a *CNN+RNN+Dense_sep_MFCC* hálózat hasonló eredményt hozott.

Ezzel ellentétben a *CNN+RNN+Dense_sep* hálózat valamelyest alulmradt, 11%-kal rosszabb eredményt hozott. Ez azonban még mindig biztató eredmény figyelembe véve, hogy nem előfeldolgozott adatokkal dolgozott.

Legrossazbbul ezúttal is az *LSTM+Dense* hálózat teljesített. Ez a pontosság messze nem elegendő bármilyen felhasználásra.

E. Tévesztési mátrixok

A CNN+Dense_sep modell tévesztési mátrixát 1. ábra mutatja. Jól látható, hogy a tévesztések jelentős része három szóhoz tartozik és ezeket a szavakat egy konkrét szóval kever össze legtöbbször. Ez a hiba valószínűleg a szópárok hangtani hasonlóságából származik. Összeségében azonban pozítívan értékelhető az eredmény.

A CNN+Dense modell tévesztési mátrixát mutatja be a 2. ábra. Kiugró, hogy nem ugyan annyiszor fordult elő minden szó, de ettől eltekintve jól értelmezhető marad a táblázat. A CNN+Dense_sep modellhez képest megfigyelhető, hogy több olyan szó van, amelyeknél jelentősen alacsonyabb a helyes predikciók száma. Ez egy sokkal szerencsétlenebb eset, mert több szóban is kevésbé biztos, ami jelentősen csökkenti a megbízhatóság megítélését.

3. ábra mutatja az LSTM+Dense modell tévesztési mátrixát. Az eddigi eredményeknek megfelelően, jelentős alul marad a 2D konvolúció alapú hálózatokhoz képest. Érdemes azonban megjegyezni, hogy a legnagyobb értékek ebben az esetben is a főátlóban vannak. Ez mellett azonban az is megfigyelhető, hogy bizonyos szavakat jelentősen többször prediktált a hálózat, tehát néhány kiemelt szót részesít előnyben.

4. ábrán látható a CNN+RNN+Dense_sep_MFCC modell tévesztési mátrixa. Itt is az figyelhető meg, mint a 2. ábrán, a szavaknak nagyjából a felét jelentősen mgasabb arányban téveszti el és jellemzően egy bizonyos szót prediktál helyettük.

5. ábra mutatja a CNN+RNN+Dense_sep hálózat tévesztési mátrixát. Az előzőhöz hasonló eredményt tapasztalunk azzal a különbséggel, hogy a helyes predikciók száma a gyakran elrontott szavaknál alacsonyabb, mint az előző esetben és a hibás predikciók is jobban szétoszlanak a többi szó között.

VI. ÖSSZEGZÉS, KONKLÚZIÓ

Dolgozatom során több típusú neurális hálózatot tanítottam be több féle adatelőkészítéssel kis szótáras hangfelismerő feladatra. Célom egy olyan struktúra keresése volt, amely aklamas robot hangvezérlő

rendszerként funkcionálni, tehát nincs nagy memória és számítási kpacitás igénye, valamint megbízható pontosságot hoz. Ennek megfelelően pontosság, precizitás, felidézés, f1 mérőszám, kereszt-entrópia költség, paraméterszám és tévesztési mátrix alapján hasonlítottam össze az egyes modelleket. A modellek egy része 2D konvolúció alapú, egy másik részük 1D konvolúció és LSTM alapú, valamint egy hálózat tisztán LSTM alapú volt. Az eredmények alapján egyértelműen a 2D konvolúciót alapú, MFCC-vel előkészített adatsorral dolgozó hálózat bizonyult a legjobbnak, ugyanis viszonylagosan alacsony a paraméterszáma, a legmagasabb a pontossága és precizitása, valamint a tévesztési mátrixa alapján kevés szónál jellemző a gyakori tévesztés és akkor is meghatározhatóak jellemző szópárok, amiket könnyen összekever.

VII. Elismerés

A szerző ez úton is szeretné megköszönni minden önkéntes munkáját, aki segített a felvételek elkészítésében, valamint Szaszák György tanár úr folyamatos iránymutatását a feladat megoldás közben.

VIII. Hivatkozások

[1] A.-r. M. a. G. H. Alex Graves, “SPEECH RECOGNITION WITH DEEP RECURRENT NEURAL NETWORKS,” in *IEEE*, Vancouver, BC, Canada, 2013.

[2] A. S. F. B. Has¸im Sak, „Long Short-Term Memory Recurrent Neural Network Architectures for Large Scale Acoustic Modeling,” in *International Speech Communication Association*, Singapore, 2014.

[3] L. D. D. Y. Ossama Abdel-Hamid, “Exploring Convolutional Neural Network Structures and Optimization Techniques for Speech Recognition,” 2013.

[4] A.-r. M. B. K. Tara N. Sainath, “Deep convolutional neural networks for LVCSR,” in *IEEE*, Vancouver, BC, 2013.

[5] X. W. Tong Fu, „Multi-scale feature based convolutional neural networks for large vocabulary speech recognition,” in *IEEE*, Hong Kong, China, 2017.

[6] R. A. E. B. C. C. J. C. B. C. Dario Amodei, “END-TO-END SPEECH RECOGNITION IN ENGLISH AND MANDARIN,” in *ICLR*, Caribe Hilton, San Juan, Puerto Rico, 2016.

[7] N. J. Alex Graves, “Towards End-to-End Speech Recognition with Recurrent Neural Networks,” in *International Conference on Machine Learning*, Beijing, China, 2014.

[8] A.-r. M. H. J. L. D. G. P. D. Y. Ossama Abdel-Hamid, “Convolutional Neural Networks for Speech Recognition,” *AUDIO, SPEECH, AND LANGUAGE PROCESSING*, vol. 22, no. 10, p. 1533, 2014.

[9] W. X. W. G. Du Guiming, “Speech recognition based on convolutional neural networks,” in *IEEE*, Beijing, China, 2016.

[10] A. S. F. B. Has¸im Sak, „LONG SHORT-TERM MEMORY BASED RECURRENT NEURAL NETWORK ARCHITECTURES FOR LARGE VOCABULARY SPEECH RECOGNITION,” in *IEEE*, FLORENCE, ITALY, 2014.