Mély neuronhálós beszédfelismerők működésének értelmező elemzése

Grósz Tamás, Tóth László

Számítógépes Algoritmusok és Mesterséges Intelligencia Tanszék
Szegedi Tudományegyetem, Informatikai Intézet
Szeged, Árpád tér 2.
{groszt, tothl}@inf.u-szeged.hu

Kivonat Manapság nyilvánvalóvá vált, hogy beszédfelismerésben a mély neuronhálós modellek teljesítenek a legjobban, azonban fontos kérdés, hogy miért működnek ilyen jól. Az utóbbi pár évben megnövekedett a igény, hogy a mély hálókat ne csupán fekete dobozként kezeljük, hanem azok belső működését próbáljuk megérteni, interpretálni is. Az interpretálásra több eszköz is létezik, jelen cikkben mi két beágyazási technikát alkalmazunk annak vizsgálatára, hogy egy neuronhálós beszédfelismerőn belül pontosan mi történik használat közben. A vizsgált háló egy magyar nyelvű beszédfelismerő része, amelyet egy híradós adatbázison tanítottunk. A háló struktúráját tekintve nem rendelkezik könnyen értelmezhető, keskeny üvegnyak (bottleneck) réteggel, ezért a neuronháló nagy méretű rejtett rétegeinek kimeneteit tanulmányoztuk. Első vizsgálataink során arra a kérdésre kerestük a választ, hogy mennyire jól különíti el az adott réteg a magán- és mássalhangzókat, valamint a csendes részeket. A következő lépésben azt tanulmányoztuk, hogy a magán- és mássalhangzókon belül más csoportok reprezentációja is azonosítható-e. Eredményeink alapján megállapítható, hogy a mély háló számos olyan tulajdonságot is megtanult a beszédhangokról, amelyek felismerésére explicit módon nem tanítottuk a hálót.
Kulcsszavak: mély neuronhálók, interpretálhatóság, beszédfelismerés

1. Bevezetés

Az elmúlt pár évben egyértelművé vált, hogy a mély neuronhálós beszédfelismerők sokkal jobb eredményeket tudnak elérni, mint más technikák [1]. Megjelenésük óta főleg a technológia finomítására fókuszált a beszédfeldolgozó közösség, minél jobb eredmények elérése céljából és kevésbé törődtek annak a fontos kérdésnek a megválaszolásával, hogy miért is működnek ilyen jól a mély neuronhálók beszédfelismerésben. Ez a trend változni látszik; a közelmúltban több tanulmány is megjelent, amelyek a beszédfelismerőkben található hálók működését elemzik és az interpretálhatóság javítását célozzák [2,3,4,5,6].

Az interpretálhatóság még nem egy teljesen kiforrott tématerület, ám egyre fontosabbá válik, ahogy a mesterséges intelligencia mindennapjaink részévé válik, hiszen az emberek többsége nehezen bízik meg egy olyan rendszerben, amit

nem ért, nem tudja miért működik. Egy betanított modell értelmezésére többféle módszer is létezik; globális vizsgálat esetén magát a modellt próbáljuk értelmezni, míg lokális esetben egy adott bemenethez tartozó kimenetekhez keresünk magyarázatot [7]. Jelen munkában mi ez utóbbira fókuszálunk, azaz azt próbáljuk megmutatni, hogy adott bemenet esetén mi történik a hálózat belsejében. A lokális értelmezés egyik fő eszköze a rejtett rétegek aktivációinak vizualizálása, ehhez viszont át kell transzformálni az általában magas dimenziószámú vektorokat alacsonyabb (általában kettő) dimenziós térbe, hogy emberek számára is átlátható legyen. Ezt a transzformációt dimenzióredukciós módszerekkel tudjuk elvégezni, amelyekből rengeteg létezik. Ezek közül mi két módszert alkalmaztunk vizsgálataink során: a neuronhálókhoz javasolt t-sztochasztikus szomszéd beágyazása (t-Stochastic Neighbor Embedding, t-SNE) [8] és a közelmúltban javasolt egyenletes sokaság becslése és projekciója (Uniform Manifold Approximation and Projection, UMAP) módszert [9].

A korábbi művekben [3,6] speciális neuronháló struktúrát használtak, úgynevezett üvegnyak (bottleneck) réteget alkalmazva. Ez lényegében egy, a háló többi rétegéhez képest kevesebb neuront tartalmazó rejtett réteg, ezen szűk rétegnek a kimeneteit könnyen lehet vizsgálni különböző beágyazási technikákkal. Mi ezzel ellentétben egy már korábban betanított háló működésének elemzését tűztük ki célként, így nem alkalmaztunk szűkített rejtett réteget. Vizsgálataink során két népszerű beágyazási technika segítségével vizsgáltuk meg, hogy egy jól működő magyar nyelvű beszédfelismerő neuronhálója pontosan hogyan is működik. A hálónk egy 5 rejtett réteges háló volt, minden rejtett rétegben 1000 ReLU neuron található (struktúrája és tanítási paraméterei megegyeznek a [10] műben leírtakkal). A neuronháló tanításához egy magyar nyelvű híradós adatbázist [11] használtunk. Az interpretálhatóság céljából kiértékeltük a hálót egy kellően hosszú hangfájlon, amelyet a teszt halmazból választottunk, majd több rejtett réteg kimenetét is beágyaztuk a kettő dimenziós térbe, hogy vizualizálhassuk, milyen belső reprezentációk (fonémakategóriák) alakultak ki a hálóban.

2. Beágyazási technikák

Ahogy korábban említettük, több beágyazási technika is létezik. Jelen munkában, hogy biztosan ne vonjunk le téves következtetéseket egyetlen módszer eredményei alapján, két lehetséges technikára fókuszáltunk. Az első módszer, a t-SNE algoritmus [8] eredetileg is mély hálókban található rejtett rétegek kimeneteinek transzformálására lett javasolva, illetve az UMAP beágyazás [9], amely a t-SNE egyik legújabb alternatívája. A továbbiakban röviden bemutatjuk ezen két módszert.

2.1. T-SNE

A t-SNE egy felügyelet nélküli módszer, amelynek segítségével mély hálók rejtett rétegeinek kimeneti értékeit ágyazhatjuk be alacsony dimenziós térbe [8]. Ezen

beágyazás segítségével vizualizálhatjuk a háló belső működését annak interpretálása céljából.

A módszer maga tekinthető dimenzióredukciós módszernek, amelynek célja, hogy a lehető legtöbbet megőrizzen a magas dimenziós struktúrából miközben áttranszformálja az adatot egy lényegesen alacsonyabb dimenziós térbe. Esetünkben a rejtett rétegek kimenetei 1000 dimenziós vektorokat generáltak, amelyeket vizualizálás céljából kettő dimenziós síkra redukálunk.

A t-SNE algoritmus két fontos lépésből áll. Az első lépés során a magas dimenziós térben az adatpontok közötti euklideszi távolságot alakítja át feltételes valószínűségekké, amelyek a pontok közötti hasonlóságot fogják reprezentálni. A második szakaszban maga a beágyazás történik, a pontok elhelyezése az alacsonyabb dimenziós térben. Ezt egy optimalizáló algoritmus végzi el, a korábban kiszámolt hasonlóságok alapján.

Tekintsük első körben meg, hogyan pontosan hogyan számolható hasonlóság két pont között magas dimenzióban a t-SNE módszer segítségével. Tegyük fel, hogy x_i és x_j két pont az N-dimenziós térben, ekkor a módszer első lépésben egy feltételes valószínűséget ($p_{j|i}$) definiál:

$$p_{j|i} = \frac{exp(-||x_i - x_j||^2/2\sigma_i^2))}{\sum_{k \neq i} exp(-||x_i - x_k||^2/2\sigma_i^2)}. \tag{1}$$

Ez a valószínűség a szerzők szerint úgy értelmezhető, hogy mekkora a valószínűsége annak, hogy x_i pont az x_j-t választja szomszédjának, amennyiben a szomszédok kiválasztásának valószínűsége arányos egy x_i középpontú Gauss eloszlással, aminek szórása a σ_i^2. A szórások beállítását a felező módszerrel tudjuk elvégezni úgy, hogy a feltételes eloszlások perplexitása egy előre megadott értéknek feleljen meg, ezzel tudjuk elérni, hogy a tér sűrűbb részeiben kisebb σ_i^2 értékek lesznek. A hasonlóságot a pontok között N dimenzióban a $p_{j|i}$ valószínűségek alapján számolhatjuk:

$$d_{ij} = \frac{p_{j|i} + p_{i|j}}{2N}, \tag{2}$$

és $i = j$ esetén $d_{ij} = 0$.

Maga a transzformáció alacsonyabb (D) térbe egy optimalizálási problémának tekinthető, amihez első lépésben definiálnunk kell egy hasonlóságfüggvényt a D dimenziós térben is. Ezen függvénnyel próbáljuk mérni a hasonlóságot a x_i és x_j pontok transzformáltja, az y_i és y_j pontok között:

$$q_{ij} = \frac{(1 + ||y_i - y_j||^2)^{-1}}{\sum_{k \neq l}(1 + ||y_k - y_l||^2)^{-1}}, \tag{3}$$

amennyiben $i = j$, akkor $q_{ij} = 0$. A képletből látható, hogy 1 szabadsági fokú Student-féle t-eloszlást (más néven Cauchy eloszlás) használ a módszer, aminek hasznos tulajdonsága, hogy a távoli pontok beágyazása majdnem teljesen invariáns lesz a tér átskálázására, illetve távoli klaszterek pontjai hasonló módon befolyásolják egy pont elhelyezkedését, mint ha különálló pontok lennének. Ez utóbbi tulajdonság az optimalizáló számára lesz hasznos.

Végül az y_i pontok elhelyezéséhez iteratív módon a következő Kullback-Leibler divergenciát minimalizáljuk:

$$KL(P||Q) = \sum_{i \neq j} d_{ij} log \frac{d_{ij}}{q_{ij}}. \quad (4)$$

Ez a módszer az egyik legszélesebb körben elterjedt technika rejtett rétegek aktivációinak vizualizálására és elemzésére, számos területen alkalmazták már pl. képfeldolgozásban [12], természetes nyelvi feldolgozásban [13] és beszédfelismerésben [6]. Hátránya, hogy számos paramétert (perplexitás, optimalizálási iterációk száma, stb.) kell megfelelően beállítanunk ahhoz, hogy jól működjön.

2.2. UMAP beágyazás

Az UMAP módszer megértéséhez fontos ismernünk a sokaság (manifold) fogalmát, amit röviden úgy lehet jellemezni, hogy egy olyan topológiai tér, amely lokálisan minden pont környezetében homeomorf a megfelelő dimenziós Euklideszi tér egy-egy nyílt halmazával [14]. A módszer három fontos feltételezésen alapszik:

- az adat egyenletesen oszlik el egy Riemann sokaságon,
- a Riemann metrika lokálisan konstans (vagy becsülhető úgy),
- a sokaság lokálisan összefüggő.

Ezen feltevések alapján az algoritmus első lépésben egy sokaságot keres, amelyen a magas dimenziós adat közel egyenletesen oszlik el, ami természetesen valós adat esetén nem feltétlenül teljesül. A probléma megoldására egy Riemann metrikát kell keresnünk, aminek használata esetén teljesül, hogy a pontok egyenletesen oszlanak el a sokaságon. Ezen Riemann metrika használatával lényegében különböző távolságokat használunk minden pont esetén lokálisan és ezen távolságok nem feltétlenül lesznek kompatibilisek. Következő lépésben a módszer ezeket az inkompatibilis lokális adatokat a sokaságon egyesíti majd átalakítja egy fuzzy topológiai reprezentációvá.

A beágyazást itt is egy optimalizálási problémamegoldásával végezzük el, mégpedig úgy, hogy az alacsonyabb dimenzióban elhelyezett pontokhoz is kinyerjük azoknak a topológiai reprezentációját (hasonló módon mint a magas dimenzió esetén) és a két fuzzy topológiai reprezentáció kereszt-entrópiáját minimalizáljuk a beágyazott pontok átmozgatásával. A módszer részletesebben az eredeti műben [9] kerül bemutatásra a matematikai háttérrel együtt.

Az UMAP módszer 2018-ban jelent meg, így még nem terjedt el olyan széles körben, mint a t-SNE, de használata több szempontból is előnyösebb. Talán a legfontosabb tulajdonsága, hogy lényegesen gyorsabban működik mint a t-SNE nagy méretű és magas dimenziós adatbázisok esetén. A sebességen túl a szerzők szerint az UMAP jobban megőrzi az adatban található globális struktúrát mint a t-SNE módszer [9], ez utóbbi állítást a mi kísérleteink is igazolták.

Csoport	fonémák
magánhangzók	
mély hangrendű	a, á, u, ú, o, ó
magas hangrendű	e, é, i, í, ö, ő, ü, ű
mássalhangzók	
zárhangok	p, b, t, d, k, g, ty, gy
réshangok	f, v, s, sz, z, zs, h
zárréshangok	c, cs, dz, dzs
nazális hangok	m, n, ny
egyéb	l, ly, r, j

1. táblázat. A vizsgálataink során használt beszédhang-kategóriák.

3. Beszédhang-kategóriák

Az adatokon végzett dimenzióredukció után fontos, hogy megvizsgáljuk, milyen klaszterek alakultak ki. Ehhez első lépésben 3 kategória elkülönülését vizsgáltuk, a magán- és mássalhangzók mellett a csend kategóriába soroltuk azokat a részeket, ahol nem volt beszéd,valamint a zárhangok (closure) szakaszait is. Ezen szinten főleg arra voltunk kíváncsiak, hogy mennyire különülnek el a magán- és mássalhangzók egymástól, hiszen a csendes részeket elég nagy pontossággal felismerte a rendszer, így azt valószínűleg jól elkülönítette a másik két csoporttól. A következő lépésben a magán- és mássalhangzókat osztottuk további kategóriákra, a magánhangzókat hangrend szerint, a mássalhangzókat pedig a képzés módja szerint, remélve, hogy a neuronháló is valami hasonló belső felosztást alakított ki anélkül, hogy erre külön tanítottuk volna. A kialakított csoportokat az 1. táblázat foglalja össze.

4. Eredmények

A kísérleteink során a teszthalmazból kiválasztottunk egy hangfájlt, amelyhez a flat-start során használt rendszerünkkel készítettünk kényszerített illesztéssel időben illesztett címkéket. A következő lépésben kiértékeltük a mély hálónkat a hangfájlon és elmentettük a rejtett rétegek kimeneti értékeit. A beágyazás során a t-SNE esetén az első rejtett réteg kimeneteit felhasználva, a beágyazás minőségét vizuálisan értékelve állítottuk be a módszer paramétereit (a perplexitást 50-re, az iterációszámot pedig 5000-re). A továbbiakban is ezeket az értékeket használtuk. UMAP esetén könnyebb volt a helyzetünk, mivel az alapértelmezett paraméterekkel is jól működött az algoritmus, nem volt szükség azok beállítására. Tapasztalataink alapján az UMAP futtatása nagyjából negyed annyi időt igényelt, mint a t-SNE.

Első lépésben megvizsgáltuk, hogy a kimeneti vektoraink mennyire ritkák, hiszen az ismert, hogy ReLU aktivációs függvény használata esetén a neuronok jelentős része inaktív lesz, tehát nullát ad kimenetként. Megfigyelhető, hogy a

Rejtett réteg sorszáma	Aktivitás
1	35.0%
2	27.6%
3	24.9%
4	21.9%
5	25.6%

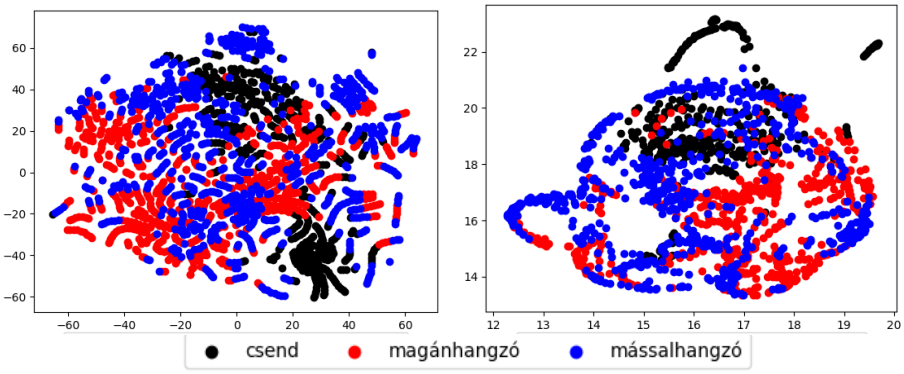
2. táblázat. A rejtett rétegekben az aktív (nem 0 kimenetet adó) neuronok aránya, a rétegek sorszámozása a bemenet felől a kimenet felé növekszik.

legnagyobb aktivitás a bemenetet figyelő rejtett rétegben volt, a neuronok közel 35%-a volt aktív. Érdekesség, hogy a kimenet felé haladva a magasabb rejtett rétegekben az aktív egységek száma csökken, azaz egyre kevesebb neuronnal nyerünk ki hasznos információt, de a kimeneti réteg alatti rétegben hirtelen megnövekszik a nem nulla kimenetek aránya. Véleményünk szerint a magyarázat az lehet erre, hogy a kimeneti réteg ezen réteg kimeneteire támaszkodva hoz döntést, ezért szükséges nagyobb arányú aktivitás. Ezen hipotézisünk igazolásához további vizsgálatok lennének szükségesek, hogy megvizsgáljuk vajon ez a jelenség más rejtett réteg-szám esetén is jelentkezik-e.

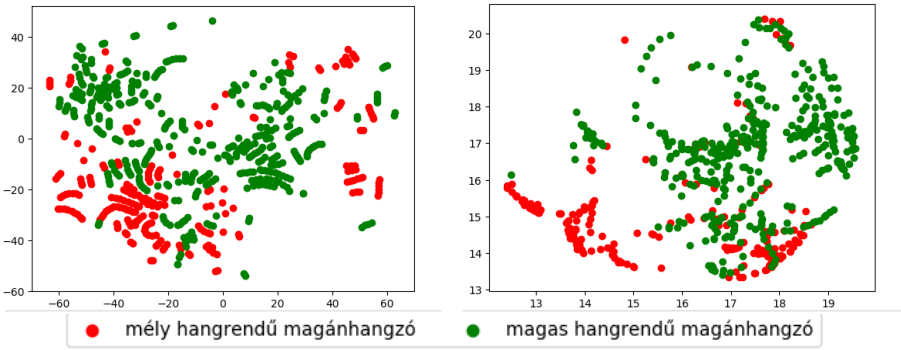
Miután megvizsgáltuk a rétegek aktivitását, figyelmünket a két fontos rétegre fókuszáljuk; a bemeneti réteghez csatolt első rejtett rétegre valamint a kimeneti réteg által figyelt utolsó rejtett rétegre. Tekintsük meg először, hogy egész pontosan milyen kimeneteket generált a legelső rejtett réteg, azaz milyen alacsony szintű jellemzőket nyert ki a bemenetből, azok mennyire jól szeparálják a korábban ismertetett beszédhang-kategóriákat. Első lépésben tekintsük az 1. ábrát, amelyen minden adatkerethez beágyaztuk kettő dimenzióba az első rejtett réteg kimenetét, majd az időben illesztett címkéink alapján minden ponthoz egy kategóriát rendeltünk. Megállapíthatjuk, hogy két csend klaszter alakult ki, az egyik a bemondás elején, végén, illetve a szavak között hallható csendnek felel meg, míg a másik klaszter a szavakban előforduló zár (closure), ez utóbbit a mássalhangzókkal keverve láthatjuk az ábrán. Fontos megemlíteni, hogy az ábrákon láthatunk majd 1-1 kiugró pontot, amely más kategóriák klasztereibe keveredett, ezek általában a fonémahatárok környékére eső kimenetek, ahol a címke bizonytalan, hiszen az időbeli illesztést egy másik háló végezte. Ezt a jelenséget tovább erősítette a tény, hogy három állapotú fonémamodellt használtunk, azaz feltételezzük, hogy minden hang legalább 3 keret hosszú, ami a valóságban nem mindig teljesül.

A magán- és mássalhangzókkal kapcsolatban azt állapíthatjuk meg, hogy ugyan nem teljesen elkülöníthetőek két dimenzióban, de itt is kialakultak csoportok. A továbbiakban ezeket elemezzük alaposabban.

A magánhangzókat tovább vizsgálva a 2. ábrán láthatjuk, hogy már elkezdődött a magas és mély hangrendűek különválasztása, azonban ez még nem tökéletes.



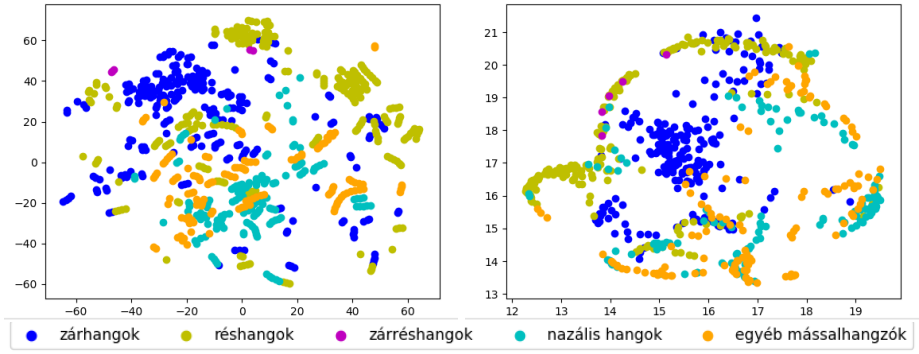
1. ábra: Az első rejtett réteg kimenetének beágyazása, balra a t-sne, jobbra pedig az UMAP módszerrel.



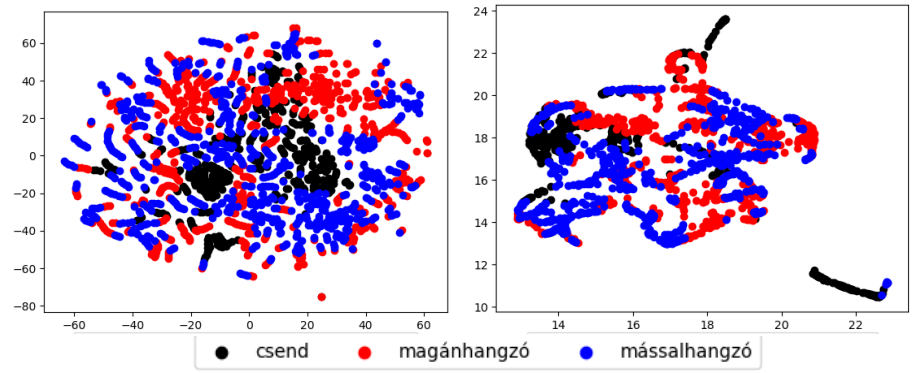
2. ábra: A magánhangzók kategorizálása az első rejtett réteg alapján, balra a t-sne, jobbra pedig az UMAP módszerrel.

Mássalhangzók esetén jól látható a 3. ábrán, hogy a zár- és réshangok elkülönülnek egymástól, azonban a többi kategória nem igazán van megkülönböztetve a háló által. Érdekesség, hogy a réshangok esetén két külön klaszter látszódik kialakulni, t-SNE esetén jól láthatóan, UMAP esetén kevésbé látványosan, de ott is látható egy szakadás a sárga klaszterben a (15,20) pont környékén. Tovább elemezve ezen két csoportot megállapítottuk, hogy az egyikben főleg zöngés, a másikban pedig zöngétlen réshangok találhatóak, tehát a háló erre vonatkozó információt is kinyert.

A legmagasabb szintű jellemzőket kinyerő réteget vizsgálva (4. ábra) látható, hogy az első réteghez hasonló módon itt sem különülnek el markánsan a magán- és mássalhangzók, de a csendes részeket itt három részre bontotta a háló, ismét megkülönböztetve a csendet a zártól. A két elkülönülő csoport közül a t-SNE esetén a nagyobb rész (a (-15,-15) környékén lévő klaszter) a szavak közötti csendnek felelt meg, a (-10,-45) körüli pedig a felvétel elején és végén hallható



3. ábra: A mássalhangzók kategorizálása az első rejtett réteg alapján, balra a t-sne, jobbra pedig az UMAP módszerrel.

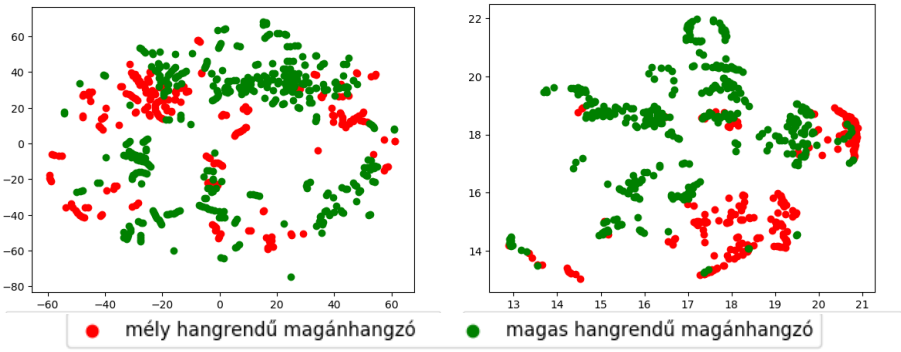


4. ábra: Az legfelső rejtett réteg kimenetének beágyazása, balra a t-SNE, jobbra pedig az UMAP módszerrel.

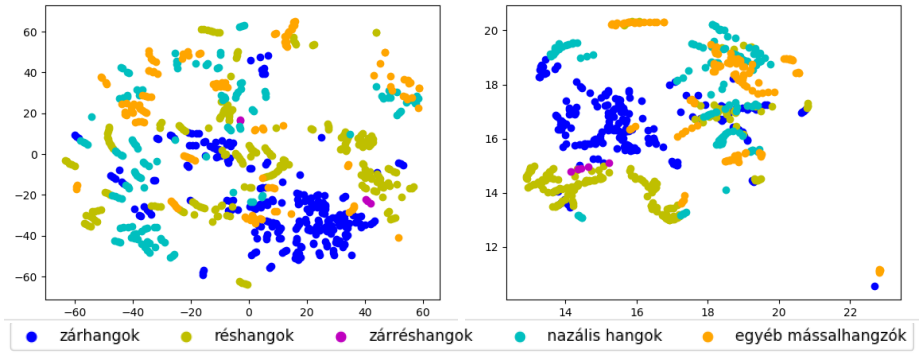
csend. UMAP esetén a két kinyúló rész közül a felső felvétel elején és végén lévő csendes rész, az alsó elkülönülő rész pedig a szavak közötti csend. Az elkülönülés már az első réteg kimeneti esetén is elkezdődött, de nem volt ennyire látványos. Ezek alapján megállapíthatjuk, hogy ez a réteg nem csupán felismeri a csendet, hanem különbséget tesz a hosszabb csend és a szavak közötti rövidebb csend között is.

Magánhangzók esetén azt láthatjuk a 5. ábrán, hogy míg UMAP alapján elég jól elkülönültek a magas és mély hangok, a t-SNE módszer esetén ez kevésbé látható. Ennek egy lehetséges magyarázata, hogy a t-SNE esetén a paramétereket újra be kellett volna állítani a jobb működés érdekében, és lehetséges, hogy nem az optimális értékeket választottuk.

A 6. ábrán a mássalhangzókhoz tartozó kimenetek beágyazása látható, az első rejtett réteghez hasonlóan itt is jól elkülönülnek a rés- és zárhangok, illetve



5. ábra: A magánhangzók kategorizálása a legfelső rejtett réteg alapján, balra a t-SNE, jobbra pedig az UMAP módszerrel.



6. ábra: A mássalhangzók kategorizálása a legfelső rejtett réteg alapján, balra a t-SNE, jobbra pedig az UMAP módszerrel.

a zárréshangok klasztere a kettő közé kerül. Az UMAP módszerrel ismét látható, hogy kialakul a zöngés és zöngétlen zárhangok csoportja, amelyek ezen rétegben már sokkal sűrűbben helyezkednek el. Tekintve, hogy a neuronháló ezen rétege se igazán tesz különbséget a nazális és egyéb magánhangzók között kijelenthetjük, hogy a beszédfelismerő ilyen jellegű információt nem tanult meg kinyerni a tanító adatból.

5. Összegzés

Munkánk során egy magyar nyelvű beszédfelismerő mély neuronhálós modulját elemeztük interpretálhatóság céljából. A hálót kiértékeltük egy teszt hangfájlon, majd a kapott rejtett rétegek kimeneteit vizsgáltuk meg alaposabban. A legelső és legfelső rejtett rétegek aktivációs értékeit két beágyazási módszerrel (t-SNE és

UMAP) levetítettük kettő dimenziós térbe, hogy ábrázolhassuk azokat elemzés céljából.

A kapott beágyazások alapján megállapítható, hogy a háló már alacsonyabb rétegeiben is elkezdte különválasztani a csendes részeket a beszédet tartalmazó résztől, illetve megkülönböztette a zárt és a valódi csendet. Magasabb szinten pedig már a szavak közötti csendet is elkülönítette a felvétel elején és végén hallható csendtől. A magánhangzók esetén a legfelső rétegben a magas és mély hangrendű hangok megkülönböztetését is megfigyelhetjük. Mássalhangzókat tekintve két fontos csoportot tanult meg felismerni a háló, mégpedig a zár és a réshangokat, utóbbi esetén még a zöngésséget is figyelembe vette a neuronháló. Az eredményeink alapján megállapítható, hogy a beszédfelismerő számos olyan dolgot is megtanult, amit explicit módon nem vártunk el tőle.

Köszönetnyilvánítás

Grósz Tamás munkáját a Nemzeti Kutatási, Fejlesztési és Innovációs Hivatal Mesterséges Intelligencia Nemzeti Kiválósági Programja támogatta a 2018-1.2.1-NKP-2018-00008 azonosítójú projekt keretében.

Tóth Lászlót az MTA Bolyai János Kutatási Ösztöndíja, valamint az Emberi Erőforrások Minisztériuma ÚNKP-18-4 kódszámú Új Nemzeti Kiválóság Programja támogatta.

A kutatást az Emberi Erőforrások Minisztériuma Emberi Erőforrások Minisztériuma 20391-3/2018/FEKUSTRAT kódjelű pályázata támogatta. A kutatáshoz használt grafikus kártyát az NVIDIA Corporation ajándékozta csoportunknak.

Hivatkozások

1. Hinton, G., Deng, L., Yu, D., Dahl, G.E., Mohamed, A.r., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T.N., et al.: Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. IEEE Signal processing magazine **29**(6) (2012) 82–97
2. Mohamed, A., Hinton, G., Penn, G.: Understanding how deep belief networks perform acoustic modelling. In: Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). (2012) 4273–4276
3. Vu, N.T., Weiner, J., Schultz, T.: Investigating the learning effect of multilingual bottle-neck features for ASR. In: Proc. Interspeech. (2014) Interspeech 2014.
4. Tan, S., Sim, K.C., Gales, M.: Improving the interpretability of deep neural networks with stimulated learning. In: 2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU). (2015) 617–623
5. Nagamine, T., Seltzer, M.L., Mesgarani, N.: Exploring how deep neural networks form phonemic categories. In: INTERSPEECH. (2015)
6. Bai, L., Weber, P., Jančovič, P., Russell, M.: Exploring how phone classification neural networks learn phonetic information by visualising and interpreting bottleneck features. In: Proc. Interspeech. (2018) 1472–1476

7. Lipton, Z.C.: The mythos of model interpretability. ACM Queue **16**(3) (2018) 30:31–30:57
8. van der Maaten, L., Hinton, G.: Visualizing data using t-sne. Journal of machine learning research **9**(Nov) (2008) 2579–2605
9. McInnes, L., Healy, J.: Umap: Uniform manifold approximation and projection for dimension reduction. arXiv preprint arXiv:1802.03426 (2018)
10. Grósz, T.: Training Methods for Deep Neural Network-Based Acoustic Models in Speech Recognition. PhD thesis (2018)
11. Tóth, L., Grósz, T.: A comparison of deep neural network training methods for large vocabulary speech recognition. In: Proceedings of TSD. (2013) 36–43
12. Esteva, A., Kuprel, B., Novoa, R.A., Ko, J., Swetter, S.M., Blau, H.M., Thrun, S.: Dermatologist-level classification of skin cancer with deep neural networks. Nature **542**(7639) (2017) 115
13. Narasimhan, K., Kulkarni, T., Barzilay, R.: Language understanding for text-based games using deep reinforcement learning. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics (2015) 1–11
14. Lee, J.M.: Riemannian manifolds: an introduction to curvature. Volume 176. Springer Science & Business Media (2006)