

1. Az első két év eredményeinek összefoglalója

Orvosi leletek gyűjtése, feldolgozása:

Az adatgyűjtés során klinikai partnereinktől nagy mennyiségű orvosi leletanyagot gyűjtöttünk be, mind az endoszkópiai, mind az ultrahangos leletezés területéről. A begyűjtött anyagokat rendszereztünk, egységes formátumra hoztuk, a helyesírási hibákat korrigáltuk, az előforduló szóalakok eloszlását statisztikailag elemeztük.

A Magyar Referencia Beszédadatbázist (MRBA) létrehozása:

Megterveztük egy beszédadatbázis szöveganyagát oly módon, hogy annak tartalmi kialakítása megfeleljen a vonatkozó európai szabványoknak. Az adatbázis fő része a felismerők akusztikus modelljeinek betanítására szolgáló blokk. Az itt szereplő szöveget úgy állítottuk össze, hogy hangkapcsolatokban minél gazdagabb legyen. A Magyar Referencia Beszédadatbázist (MRBA) a BME TMIT Beszédakusztikai Laboratóriuma és a szegedi SZTE Informatikai Tanszékcsoport együttműködésben hozta létre. A cél egy olyan irodai, otthoni környezetben olvasott folyamatos szöveget tartalmazó beszédadatbázis megalkotása és akusztikai, nyelvi feldolgozása volt, amely alkalmas PC-s beszédfelismerők betanítására, tesztelésére.

A felvételek mindegyikét annotáltuk, azaz minden hangfájl mellé egy címkefájlt készítettünk, amely különféle információkat tartalmaz a hangfájl paramétereivel és tartalmával kapcsolatban: az elhangzott szöveg ortografikus lejegyzését, hibás kiejtést, nem érthető szavakat, szótöredékeket, a beszélő nem beszédből származó hangjait, környezeti zajokat, stb.

Az adatbázis közel egyharmadán, azaz 100 beszélő anyagán manuálisan fonetikai szintű szegmentálást és címkézést is végeztünk.

Nyelvi modellezés:

Az első évben a nyelvi elemzés főleg a szöveganyagok gyűjtését, elemzését jelentette. Vizsgáltuk az előforduló szavak és szóalakok számát, gyakorisági eloszlásukat. Elemeztük az adatbázisban előforduló mondatok nyelvi struktúráját. Vizsgáltuk, mennyiben lehetséges a szóhatárok automatikus kijelölése a hangsúly vizsgálata alapján.

A nyelvi betanításhoz a budapesti SOTE II. sz. Belgyógyászati Klinikájától (2700 lelet) és a szegedi Orvostudományi Egyetemről (6365 lelet) gyűjtött korábbi leletanyag korpuszt használtuk. Ezen szöveggörpusz alapján elkészítettük el a teljes szóalakszótárat, amely 14331 szót tartalmaz, a kiejtési szótárat és ezek téma szerint osztott kisebb szótárait, valamint a korpusz alapján morfémaszótárat is készítettünk, amelynek nagysága 6824 morfémaelem.

Lényegében a nyelvi modellhez n-gram modelleket használtunk, de az egyik megoldásban a hagyományos szóalakok az alkotó elemek, a másik megoldásban viszont a morféma.

A második évben a BME TMIT Beszédakusztikai Laboratóriumában kifejlesztésre került egy Windows XP alatt működő beszédfelismerő fejlesztői környezet, amely alkalmas különböző középszótáras 1000-10000 szavas szövegek betanítására és felismerésére.

A felismerő a statisztikai alapon működő HMM akusztikai fonémamodellekkel, valamint a statisztikai alapú bigram nyelvi modellel működik, nemlineáris simítást használva. Az akusztikai modelleket az MRBA beszédadatbázissal tanítottuk.

A felismerő optimális működését az akusztikai és nyelvi modellek változtatásával állítottuk be.

Külön súlyt fektettünk a valós idejű felismerés elérésére: a dinamikus címezésen és az akusztikai modellek indirekt megközelítésén túl memóriaelérési optimalizáció, valamint nyalábolt keresésnél (Beam Search) változó terű nyaláb alkalmazásával.

A vizsgálatokról konferenciaelőadások és cikkek jelentek meg.

2. A harmadik záró évben elkészült részfeladatok és az elért eredmények bemutatása

Orvosi diktálórendszerekről általában

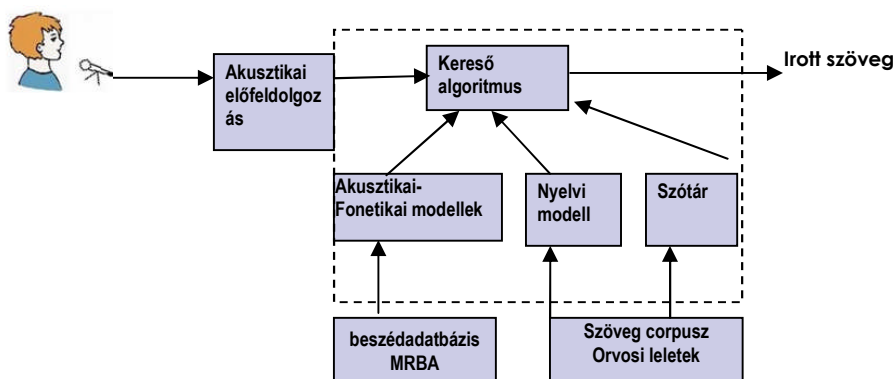
A beszéd felismerési technológia jelenleg még nem képes az általános célú folyamatos diktálás tökéletes megoldására, viszont elfogadható pontosságot tud nyújtani olyan feladatok esetében, ahol a szókinccs és a nyelvtani felépítés korlátozott. Így lehetővé teheti az ún. beszédalapú dokumentálást olyan szakmák esetében, amelyek szakszöveg-jellegű dokumentációt igényelnek. Kitűnő példa erre az orvosi vizsgálati eredmények rögzítése, amely folyamat felgyorsítása különösen nagy jelentőséggel bír. Ilyen diktáló rendszerek a világnyelvekre már léteznek, viszont kisebb nyelvekre egyelőre nagyon kevés orvosi diktáló szoftver látott napvilágot, ami többek között a nyelvi sajátosságoknak és a magas fejlesztési költségeknek tudható be.

Beszéd felismerő prototípus

Míg a második év a megoldások keresésének éve volt, a harmadik szakaszban a már kialakított rendszereket optimalizáltuk, majd véglegesítettük a felismerő prototípusát. Az elkészült prototípust klinikai partnereink tesztelték, és folyamatosan visszajelezték a tapasztalataikat illetve a felbukkanó hibákat, amelyeket korrigáltunk. Endoszkópos, valamint hasi és kismedencei ultrahangvizsgálatra fejlesztettük ki a beszéd felismerő rendszert. A „Műszaki dokumentációt”, amely tartalmazza a program működésének technikai alapjait, az 1. sz. Melléklet tartalmazza. A felismerő honlapja: <http://alpha.tmit.bme.hu/speech/research.php>.

Középszótáras endoszkópos valamint hasi és kismedencei ultrahang leletek automatikus lejegyzésére alkalmas folyamatos beszéd felismerő rendszer

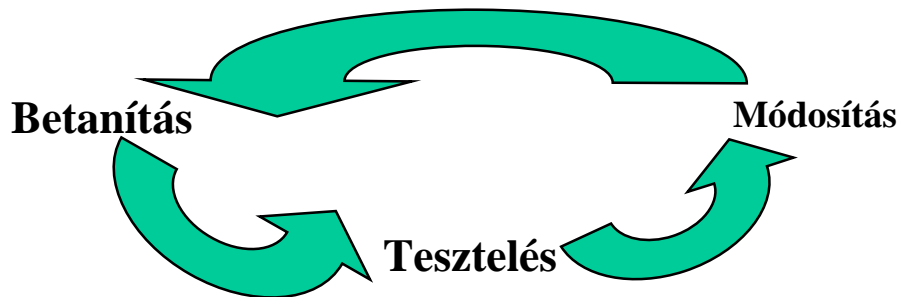
A BME TMIT Beszédaakusztikai Laboratóriumában elkészült a folyamatos beszéd felismerő rendszer, amelynek alapja a Windows XP alatt működő beszéd felismerő fejlesztői környezet, amely alkalmas különböző középszótáras 1000-10 000 szavas szövegek betanítására és felismerésére. A felismerő a statisztikai alapon működő HMM akusztikai fonémamodellekkel [1] valamint a statisztikai alapú bigram nyelvi modellel működik, nem lineáris simítást használva [2]. Az akusztikai modelleket az MRBA beszéd adatbázissal tanítottuk [5]. A nyelvi betanításhoz a budapesti SOTE II. sz. Belgyógyászati Klinikájától gyűjtött gasztroszkópiai és ultrahang leletanyag korpuszt használtuk. Ezen szövegtörzsek alapján készítettük el a teljes szóalak szótárakat, valamint a kiejtés szótárakat és ezek téma szerint osztott kisebb szótárakat. A rendszer felépítése az 1. ábrán látható.



1. ábra Beszéd felismerő blokkvázlata. **Tulajdonságok:** - beszélőfüggetlen
- HMM-alap
- folyamatos
- középszótáras
(1000-20000 szó)
- valós idejű működés

A felismerő optimalizálása

A felismerő optimális működését az akusztikai [3] és nyelvi modellek [4] változtatásával állítottuk be a 2. ábra szerint.



2. ábra A felismerő optimalizálása

Lényegében a nyelvi modellhez n-gram modelleket használtunk. Külön súlyt fektettünk a valós idejű felismerés elérésére: a dinamikus címzésen és az akusztikai modellek indirekt megközelítésén túl, memóriaelérési optimalizáció, valamint nyalábolt keresésnél (Beam Search) változó terű nyaláb alkalmazásával.

Az **akusztikai modellek betanítását** az MRBA beszéd adatbázissal végeztük [5].

Megoldandó problémák voltak:

- akusztikai paraméterek kiválasztása
 - kvantálási lépcsők számának optimális megválasztása,
 - megfelelő súlyozású simítófüggvény megválasztása,
 - a tartomány minél jobb eseménytérbeli kihasználtsága.

Végeredményben tehát a fonémaszintű felismerőnk:

16 kHz mintavételezésű,

17 Bark frekvenciatérbeli derivált

17 időbeni derivált + 17 időbeni második derivált energia bemeneti jelvektor mellett,

4-5 állapotú kvázi-folytonos, 24 lépcsős, rejtett Markov-modellekkel (QCHMM), fonéma alappal dolgozik.

Az akusztikai, fonetikai szint optimalizálásáról az első év végén beszámoltunk [3].

A **nyelvi modell előállításához** a budapesti SOTE II. sz. Belgyógyászati Klinikájától gyűjtött gasztroszkópiái (2000 lelet) és ultrahang (2970 lelet) leletanyag korpuszt használtuk. Ezen szöveg korpuszok alapján készítettük el a teljes szóalak szótárakat, amelyek a gasztroszkópiánál 14 331 szót, ultrahangnál 5922 szót tartalmaznak, valamint a kiejtés szótárakat és ezek téma szerint osztott kisebb szótárait.

Teszteléshez a gasztroszkópiái és az ultrahang vizsgálatok esetén 20 gasztroszkópiái ill. 20 ultrahang leletanyagot használtunk egyszer az orvosok, másodsor a laboratórium munkatársai bemondásában.

A tesztelés optimalizációs lépései

Tesztelési változók:

- Beszélők neme (férfi, nő, vegyes)
- Beszéd/zaj viszony
- Nyelvi modell típusa, nyelvi modell súlyaránya az akusztikai modellhez képest
- Beam search faktor nagysága (nyalábszélesség)

Mért paraméterek:

- Beszédfelismerési pontosság, szótévesztés
- Felismerési sebesség
- memóriahasználat

A Nyelvi modell leírását és a tesztelések egy részét a 2. éves jelentés tartalmazza. A paraméterek optimalizálásával a gasztroszkópiái felismerőt 25% szótévesztésről 17% szótévesztésre tudtuk javítani, Az ultrahangos felismerést pedig 8% szótévesztésre. Ezek a szótévesztések az akusztikai és nyelvi adaptációval tovább javíthatók.

Akusztikai és nyelvi adaptáció

Személyi felhasználás esetén a felismerési pontosságon jelentősen lehet javítani a felhasználó személyéhez való illeszkedéssel, vagyis a beszédfelismerő rendszerek használata előtt a felismerőt célszerű adaptálni az adott felhasználási körülményekhez, illetve felhasználóhoz kell illeszteni. Ilyen lehetőséget építettünk be a felismerő rendszerbe. Ennek egyik része az ún. akusztikai tanítás. Ekkor a rendszer nem tesz mást, mint „megtanulja” az adott felhasználó beszédének akusztikai jellegzetességeit.

Az adaptálás második, kissé bonyolultabb, de idővel könnyen elsajátítható része a nyelvi tanítás. Ekkor a beszéd-felismerő „rátanul” az adott felhasználó – azaz szakorvos – jellemző szófordulataira, beszédstílusára, ami nagymértékben javítja és gyorsítja a felismerést. A módszer hasonlóan működik ahhoz, mint amikor az asszisztens, aki már hosszabb ideje dolgozik az adott orvos mellett, sok esetben gyakorlatilag előre tudja, hogy a szakorvos hogy fog egy-egy adott mondatot megfogalmazni, azaz akár előre tudja, mit fog mondani. Ha a felismerő is megtanulja ezt, jóval kevesebb hibát fog vétetni, ehhez azonban el kell tennie egy bizonyos időnek, amíg elegendően nagy számú leletet „hall” az adott orvostól. Ehhez a rendszert tanítani kell, ezt jelenti a nyelvi tanítás. Ennek során a felhasználó már korábban lediktált, kijavított és elmentett szöveges leleteit alapul véve a rendszer félautomatikusan frissíti tudását. A tanítás azért félautomatikus, mert a felhasználónak egyrészt kijavított leleteket kell elmentenie, hogy azokat a rendszer helyesen tanulhassa meg, másrészt, ha az újonnan bemondott leletek ismeretlen szót tartalmaznak, annak kiejtését meg kell adnunk, hiszen a rendszer azt nem tudja kitalálni. Ezen túlmenően azonban minden frissítés és adaptáció (ún. nyelvi adaptáció) automatikusan zajlik, amelynek sikeres lezárulásáról a felhasználó üzenetet kap. Ezt követően a leletező rendszerbe történő újabb bejelentkezés után már az új, frissített nyelvi beállítások lesznek aktívak.

A felismerőprogram prototípusa mellett készült tesztelési jegyzőkönyv a klinikai tesztelés tapasztalatainkról. A gyakorlatban a Klinikán kapott eredményekről a gasztroszkópiai és ultrahang vizsgálatok leletezése esetén a 2. és 3. mellékletben számolunk be.

Bibliográfia

[1] Claudio Becchetti, Lucio Prina Ricotti: *Speech Recognition, Theory and C++ implementation*. Fondazione Ugo Bordoni, Rome, 1999. ISBN 0-471-97730-6

[2] Ney, H., Essen, U., Kneser, R.: *On Structuring Probabilistic Dependencies in Stochastic Language Modeling*. Computer Speech and Language, 8:1-38. oldal

[3] Velkei Szabolcs, Vicsi Klára: Beszédfelismerő modellépítési kísérletek akusztikai, fonetikai szinten, kórházi leletező beszédfelismerő kifejlesztése céljából, MSZNY 2004.

[4] Vicsi, K. - Velkei Sz. - Szaszák Gy. - Borostyán G. –Gordos Géza: Folyamatos középszótáras, beszédfelismerő rendszer fejlesztési tapasztalatai, kórházi leletező beszédfelismerő. Híradástechnika LXI. évf. (p.14-21), 2006.

[5] Vicsi Klára, Kocsor András, Teleki Csaba, Tóth László: Beszédatbázis irodai számítógép-felhasználói környezetben., II. Magyar Számítógépes Nyelvészet Konferencia, (p.315-318), 2004

1. sz. melléklet

A gasztroszkópiai és ultrahangletező beszédfelismerő leírása

Műszaki dokumentáció

1. fejezet – szerkezeti felépítés

A futtatható beszédfelismerő végleges formájában három külön programból áll össze. Ezek a következők:

1. Akusztikai Markov-modell generáló program
2. Nyelvi Bigram-mező generáló program
3. Valós idejű beszédfelismerő szoftver

Ezek a programok kihatással vannak egymás működésére: az akusztikai szint befolyásolja a felismerés minőségét, a nyelvi szint a felismerő nyelvtani bonyolultságát, míg a felismerő eredménye visszahat(hat) a nyelvi szintre. Ez lehetővé teszi a nyelvtan folyamatos bővülését, adaptálódását a megfelelő tartalomhoz.

A következőkben egyenként áttekintjük a megfelelő programok belső szerkezetét, a bennük használt matematikai modelleket és a szoftveres megvalósításait. Mivel az akusztikai modellek létrehozása (és a hozzá tartozó szoftver elkészítése) a projekt 2004-es évében megtörtént, ezért erre külön nem térünk ki.

2. fejezet – nyelvi Bigram-mező generáló program

Ebben a fejezetben áttekintjük a nyelvi motor tulajdonságait a matematikai háttértől kezdve az adatok tárolásának módjáig.

2.1. Nyelvi modell

A felismerő a nemzetközi vonalon elfogadott és használt n -gram modellt alkalmazza a nyelvi szint definiálásához. Ismeretesek ennek előnyei és hátrányai, a magyar nyelvre való illesztés problémái és a kötetlenebb nyelvtan nehézkes modellezése. Ennek ellenére a feladatban megfogalmazott jelentősen kötött szótári és nyelvtani elemek miatt mégis bigram modell megvalósítását tűztük ki célul.

Az n -gram modellekben a nyelvi modellek szószekvenciáik valószínűségének halmazából áll.

A szekvencia valószínűsége ekkor:

$$P(w_1, w_2, \dots, w_m) = P(w_1) \prod_{i=2}^m P(w_i | w_{i-1} \dots w_1)$$

A kontextust limitálva:

$$P(w_1, w_2, \dots, w_m) \cong P(w_1) \prod_{i=2}^m P(w_i | w_{i-1} \dots w_{i-n+1})$$

ahol $n > 0$ tetszőlegesen választott konstans egész. A nyelv olyan tulajdonságokkal rendelkezik, hogy a folyamat során egy későbbi állapot valószínűsége gyakorlatilag független a kezdőfeltételektől, így n értékére nem kell nagy n értéket használni. (Tipikus értékek 2-től 6-ig)

A fenti valószínűség ekkor a következőképpen számítható ki:

$$P(w_i | w_{i-1} \dots w_{i-n+1}) = \frac{N(w_i \dots w_{i-n+1})}{N(w_{i-1} \dots w_{i-n+1})}$$

ahol $N(\cdot)$ a megadott szekvencia előfordulásai száma a tanító szöveganyagban.

2.2. Programszerkezet

A nyelvi szint létrehozásához egy szekvenciális menetű segédprogram készült, melynek csupán bemenetei és kimenetei van definiálva, ezek a következők:

Bemenetek:

- *eredeti Bigram-modell*
- *szótárkészlet*
- *mintaanyag*

Kimenetek:

- *módosított Bigram-modell*
- *módosított szótárkészlet*
- *szólista*
- *az összes eddig használt mintaanyagot tartalmazó szövegfájl*

A program működése a következő:

1. *Mintaanyag beolvasása*
2. *Szótár beolvasása*
3. *Mintaanyag és szótár illesztése, a szótár bővítése*
4. *eredeti Bigram-modell beolvasása*
5. *Bigram-modell valószínűségek módosítása*
6. *Szótár mentése*
7. *Bigram-modell mentése*
8. *Mintaanyagok mentése*
9. *Szólista mentése*

A két utolsó művelet valójában szükségtelen a felismerő működése szempontjából, de hasznos információkat tartalmaz a felhasználó adminisztrátor részére.

2.3. Szótárillesztés

A szótárkészletet az aktuális mintaanyaggal össze kell vetni. Háromféle lehetőség van:

- *a szó szerepel a szótárban – nem kell tenni semmit*
- *a szó nem szerepel a szótárban – fel kell venni a szót a szótárba*
- *a szó nem szerepel a szótárban, de létezik olyan szó, mely ennek szinonímája, rövidítése – cseré definíciója*

Az első esetben értelemszerűen nem szükséges semmilyen beavatkozás.

A második esetben a megfelelő szót fel kell venni a szótárba. Ehhez kell a szó kiejtett alakja („K”), amit a felhasználónak kell megadnia. Ezután a program UnitSAMP Make(string); függvényének hívásával létrehozza a sampa átíratot. Ekkor már mindhárom szükséges adat rendelkezésre áll, így a szó felvehető a szótárba.

A harmadik esetben a felhasználónak rendelkeznie kell, hogy mely szóra cseréli az új ismeretlen szót. Ehhez a program által adott ajánlatokból tud választani egyet. Figyelem! Ha nincs a választhatók között nekünk megfelelő, akkor a szót fel kell venni! A program a választható szavakat a következőképpen választja ki:

- *az összes egy karakteres szó (rövidítés)*
- *az összes olyan szó, melynek Levehnstein távolsága kisebb, mint 2*
- *az olyan szavak, melyek csak írásjelben különböznek*

A Levehnstein távolságot az LD és LD2 függvények segítségével határozhatjuk meg.

Miután az új szavak illeszkedtek, az új mintaanyagban kicserélődik az összes csereszó. Ezt követően pedig kezdetét veszi a bigram-mező feltöltése.

2.4. Bigram-mező létrehozása

3. Adatszerkezet

A továbbiakban áttekintjük a program által használt adatok szerkezeti formáit, a fájlok struktúráit.

3.1. Szótár

A szótár tartalma a program könyvtárában elhelyezett `words.csv` fájl. Ez a fájl egy CSV formátumú adatsort tartalmaz szöveges formában. A szeparáló jel az ASCII-ban használatos függőleges | karakter. Az adathalmaz fejléce a következő: `L/K/S`, melynek jelentése: `leírt alak/kiejtett alak/sampa alak`. A leírt és kiejtett alak egyértelmű, a magyar karakterekkel helyesen- és kiejtett formában leírt szavakat tartalmazza. A sampa alak a szavak sampa karakterekkel való leírását tartalmazza, figyelembe véve, hogy a felismerő milyen sampa karaktereket alkalmaz, és azoknak mi a jelölése.

Figyelem! A felismerő által használt karakterek jelölése eltérhet a szabványos sampa jelölésrendszertől! Az illeszkedés ellenőrzése mindenkor az adminisztrátor feladata.

3.2. Bigram-mező fájl

A bigram-mező a felismerő által használt legbonyolultabb fájlstruktúra, ezért teljes részletességgel kell bemutatni.

A fájl egy tömörített adatsor, melyet a Borland Delphi `zLib` könyvtárának `streamje` segítségével tömörítettünk.

A fájl tartalma:

Elnevezés	Típus	Értelmezés
<code>Num</code>	<code>4 byte integer</code>	Szavak száma
<code>Num2</code>	<code>4 byte integer</code>	Markov-állapotok száma
<code>Size1</code>	<code>8 byte integer (int64)</code>	CS hossza
<code>CS</code>	<code>StringList</code>	Teljes tanítóanyag
<code>Size2</code>	<code>8 byte integer (int64)</code>	CS1 hossza
<code>CS1</code>	<code>StringList</code>	Szótár cserélendő szó leírt alak
<code>Size3</code>	<code>8 byte integer (int64)</code>	CS2 hossza
<code>CS2</code>	<code>StringList</code>	Szótár csereszó leírt alak
<code>Size4</code>	<code>8 byte Integer (int64)</code>	SL hossza
<code>SL</code>	<code>StringList</code>	Szavak
<code>Size5</code>	<code>8 byte Integer (int64)</code>	SL2 hossza
<code>SL2</code>	<code>StringList</code>	Sampa karakterek száma
<code>Size6</code>	<code>8 byte Integer (int64)</code>	SL3 hossza
<code>SL3</code>	<code>StringList</code>	Sampa karakter helye a Markov-bigram mezőben
<code>Size7</code>	<code>8 byte Integer (int64)</code>	SL4 hossza
<code>SL4</code>	<code>StringList</code>	Sampa alak
<code>Size8</code>	<code>8 byte Integer (int64)</code>	SL4b hossza
<code>SL4b</code>	<code>StringList</code>	Kiejtett alak
<code>Size9</code>	<code>8 byte Integer (int64)</code>	SL5 hossza
<code>SL5</code>	<code>StringList</code>	Markov-állapotok sampa nevei
<code>X1</code>	<code>4 byte Integer</code>	Bigram X1. szó
<code>Y1</code>	<code>4 byte Integer</code>	X1. szó darabszám
<code>Si1</code>	<code>Single</code>	X1. szó előfordulási valószínűség
<code>Y2</code>	<code>4 byte Integer</code>	X2. Bigram utód-átmeneteinek száma
<code>X2</code>	<code>4 byte Integer</code>	Bigram X2. szó
<code>YY</code>	<code>4 byte Integer</code>	Utód YY. Szó (sorszám)
<code>Z</code>	<code>4 byte Integer</code>	Átmenet darabszáma
<code>Si2</code>	<code>Single</code>	Átmenet előfordulási valószínűsége

A sárgával jelölt adatok Num-szor szerepelnek az adatsorban.

A zölddel jelölt adatok szintén Num-szor szerepelnek az adatsorban, azon belül a pirossal jelölt adatsorok Y2-szer szerepelnek minden zölddel jelölt adatsor után.

Ahogy az az adatszerkezetből is látszik, a Bigram-mező mátrixa egy egydimenziós listára épül. A lista minden eleméhez két további lánc csatlakozik: egyik megmondja, hogy mely állapotok következhetnek, a másik pedig, hogy milyen valószínűséggel. Ezzel az elrendezéssel csak az átmenetek számának kétszeresét kell eltárolni információként, míg a teljes mátrix az átmenetek számának négyzete. Ebből következik, hogy mindaddig hasznos memóriaszempontról ez a struktúra, amíg a modell átmeneteinek száma el nem éri a mátrix 50%-át. Jelenleg az átmenetek még az 1%-ot sem közelítik meg. (Sparse-mátrix) Az 50%-os kitöltöttség természetesen lehetetlenné teszi a felismerést már pár ezer szó esetén is, mivel a választható útvonalak száma óriási lesz.

3.3. Mintaanyag és egyéb szövegfájlok

Ezekben a fájlokban egy közös konvenció létezik: minden szó külön sorba íródik. Ezen felül a mintaanyag esetében speciális karaktereket különböztetünk meg:

- „_” „aláhúzás jel”: tiltja az előző és a következő szó között a bigram-valószínűség számlálását. Hasznos mondathatár, vagy bekezdések határának esetében, amikor nincs statisztikai jelentősége a folytatásnak, valamint a # jelek használatakor. (lásd alább)
- #+ és #- jelek: átjárhatóságot biztosítanak két független „A” és „B” mező között. A #+ jel után a „B” mező bármely tagja következhet és a „B” mező bármely tagja után következhet a #- jel. Ezzel visszatérhetünk az „A” mező megfelelő eleméhez. A #+ és #- jelek a szavakkal azonos módon működnek, de a leírt szövegben nem látszanak, akusztikai jelentésük az esetleges szünettartáson kívül nincsen.
- #1-#9 jelek: számok esetében az adott szó(szám) rangját jelöli, az akusztikában nem vesz részt, csak sorrendi okokból használatos. Egy példa: 1205=ezer#5 kettő#4 száz#3 öt#1

4. fejezet – Beszédfelismerő szoftver

1. Markov-modell

Vegyünk alapul egy N lehetséges állapotból álló rendszert. A rendszer $t=1,2..n$ diszkrét időpontokban egyik állapotból a másikba kerülhet, ahol minimum 1 időegységet (frame) tölt el. Reprezentálni a rendszert egy gráffal a következőképpen lehet: a gráf csomópontjai az állapotok, élei az állapotok közti átmenetek. Minden átmenetet engedélyezve olyan teljes gráfot kapunk, amelyben minden csomópontozhoz létezik egy olyan él, amely önmagába záródik. Mátrixokra leképezve ez azt jelenti, hogy az $N \times N$ -es mátrix minden pontjában értelmezve van egy szám. A szám nagysága a csomópontokat összekötő élék súlya, mely az állapotátmenet valószínűségét jelenti. A továbbiakban bevezetve a következő jelöléseket:

- q_t : az az állapot, amelyikben a modell a t időpillanatban van,
- a_{ij} : az i . állapotból a j . állapotba mutató él súlya, ahol $\sum_j a_{ij} = 1$,
- $\pi_i = P(q_1 = i)$ az i . állapot kezdési valószínűsége,
- valamint $O = (o_1 o_2 \dots o_T)$ a vizsgált folyamat.

Rendeljünk a csomópontokhoz valószínűségi értéket, úgy hogy az kimutassa annak valószínűségét, hogy a folyamat o_k szimbólumát éppen az adott csomópont generálta. Ehhez első megközelítésben ismerni kell a szimbólumok S halmazát. Ebben az esetben egy $B: |S| \times N$ -es mátrixal reprezentálni lehet a fenti valószínűségeket.

Ekkor Markov-modellnek nevezzük a $\lambda = (A, B, \pi)$ paraméterekkel megadott objektumot. A modell működése igen egyszerű, ezért nem térek ki rá. Röviden be kell mutatni viszont, hogy milyen módon adhatók meg a valószínűségi értékek, és hogyan kell kiszámolni azokat. A modell egy tetszőleges $q = (q_1 q_2 \dots q_T)$ állapotsorozat mentén bocsátja ki az O sorozathoz megfelelő eseményeket. Annak a valószínűsége, hogy éppen azt az eseményt generálja a modell, mint amit O alapján elvárunk:

$$P(O | q, \lambda) = \prod_{t=1}^T P(o_t | q_t, \lambda)$$

Ez átalakítások után a következő alakra hozható:

$$P(O | \lambda) = \sum_q P(O | q, \lambda) P(q | \lambda)$$

Kérdés, hogy hogyan számíthatók ki a jobb oldalon álló valószínűségek? A dinamikus programozásból ismert Forward-Backward algoritmus segítségével:

Az előre mutató valószínűségek számítása:

1. inicializáló lépés: $\alpha_1(i) = \pi_i b_i(o_1)$

2. indukciós lépés: $\alpha_{t+1}(j) = \left[\sum_{i=1}^N \alpha_t(i) a_{ij} \right] b_j(o_{t+1})$

A hátra mutató valószínűségek számítása:

1. inicializáló lépés: $\beta_T(i) = 1$

2. indukciós lépés: $\beta_t(i) = \sum_{j=1}^N a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)$

Ezekkel az algoritmusokkal egy meglévő HMM vizsgálata végezhető el. Nem esett szó azonban a modellek előállításához szükséges algoritmus(ok)ról. Ehhez rendelkezni kell olyan eseményvektorokkal, melyekhez elő szeretnénk állítani egy Markov-modellt. A továbbiakban feltételezzük K számú O eseményvektor létezését, melyek mindegyikére előállítottuk a Forward-Backward algoritmus segítségével a valószínűségi együtthatókat. Ekkor egy lokális maximumhoz vezető utat biztosít a Baum-Welch (BWA) algoritmus (Baum, 1972). Két segédváltozóra szükség lesz a számítások folyamán:

Az első jelöli annak valószínűségét, hogy O generálása során a modell a t . pillanatban az i , $t+1$. pillanatban pedig a j állapotban volt:

$$\xi_t(i, j) = P(q_t = i, q_{t+1} = j | O, \lambda)$$

$$\xi_t^k(i, j) = \frac{P(q_t = i, q_{t+1} = j, O^k | \lambda)}{P(O^k | \lambda)} = \frac{\alpha_t^k(i) a_{ij} b_j(o_{t+1}^{(k)}) \beta_{t+1}^k(j)}{\sum_{i=1}^N \sum_{j=1}^N \alpha_t^k(i) a_{ij} b_j(o_{t+1}^{(k)}) \beta_{t+1}^k(j)}$$

A második megmutatja, hogy O generálása során mekkora a valószínűsége, hogy a t . pillanatban az i . állapotban van a modell:

$$\gamma_t(i) = \sum_{j=1}^N \xi_t(i, j)$$

Ekkor az alábbi összefüggések írhatók fel:

$$\sum_{t=1}^{T-1} \gamma_t(i) = \text{az } i. \text{ állapotból induló átmenetek száma}$$

$$\sum_{t=1}^{T-1} \xi_t(i, j) = \text{az } i. \text{ állapotból } j. \text{ állapotba menő átmenetek száma}$$

Baum-Welch algoritmussal a következő iteráció paramétereit kiszámolni a következő formula szerint lehet:

kezdőállapot valószínűségek: $\bar{\pi}_j = \gamma_1(i)$

átmeneti valószínűségek: $\bar{a}_{ij} = \frac{\sum_{k=1}^K \frac{1}{P_k} \sum_{t=1}^{T_k-1} \xi_t^k(i, j)}{\sum_{k=1}^K \frac{1}{P_k} \sum_{t=1}^{T_k-1} \gamma_t^k(i)}$

kibocsátási valószínűségek: $\bar{b}_j(k) = \frac{\sum_{k=1}^K \frac{1}{P_k} \sum_{t=1}^T \gamma_t^k(j)}{\sum_{k=1}^K \frac{1}{P_k} \sum_{t=1}^T \gamma_t^k(j)}$

Ezekkel megoldottuk a véges S szimbólumkészlettel rendelkező modellek leírását. Ha az S szimbólumkészlet nem határozható meg egyértelműen, vagy S mérete túlon túl nagy (végtelen), akkor B értékeinek kiszámításához függvényeket kell értelmeznünk. Alapvetően két elfogadott módszer használatos:

- vektorkvantálás (VQHMM)
- folytonos valószínűségi tér használata (CHMM)

Az utóbbi módszer képletei hasonló felépítésűek, mint a fent megismertek. A valószínűség-sűrűség függvény ez esetben elliptikus függvények összegeként tárgyalható. Tipikus megvalósítás Gauss-eloszlások alkalmazása, melynek során a gaussi paramétereket kell beállítani a megfelelő értékre. A kibocsátási valószínűség ekkor a következő módon írható fel:

$$\bar{b}_j(o) = \sum_{k=1}^K c_{jk} N(o, \mu_{jk}, U_{jk})$$

ahol N gaussi PDF (Probability Density Function). A gauss függvény általános esetben ekkor a következő alakot ölti:

$$\bar{b}_j(o) = \frac{1}{\sqrt{(2\pi)^D \det(U_j)}} \exp\left\{-\frac{1}{2}(o - \mu_j)^T U_j (o - \mu_j)\right\}$$

Látható, hogy a szórásokat mátrixként lehet reprezentálni. Általában csak a főátlóban lévő tagokkal szokás számolni, mivel a szűrősorok paramétereinek egymás közti kovarianciája igen kicsi. (Rabiner, Juang, 1993) Ekkor a következő végleges forma jelenik meg:

$$\bar{b}_j(o) = \sum_{k=1}^M c_{jk} \frac{1}{\sqrt{2\pi} \sigma_{jk}} \exp\left\{-\frac{(o - \mu_{jk})^2}{2\sigma_{jk}^2}\right\}$$

A BWA a gaussi paraméterek kiszámításához is nyújt megoldást:

gauss-koefficiensek: $\bar{c}_{jk} = \frac{\sum_{t=1}^T \gamma_t(j, k)}{\sum_{t=1}^T \sum_{k=1}^M \gamma_t(j, k)}$

várható értékek: $\bar{\mu}_{jk} = \frac{\sum_{t=1}^T \gamma_t(j, k) o_t}{\sum_{t=1}^T \gamma_t(j, k)}$

és szórásnégyzetek: $\bar{U}_{jk} = \frac{\sum_{t=1}^T \gamma_t(j, k) (o_t - \mu_{jk})(o_t - \mu_{jk})'}{\sum_{t=1}^T \gamma_t(j, k)}$

A képletekben feltüntetett segédfüggvény pedig:

$$\bar{\gamma}_t(j, k) = \left[\frac{\alpha_t(j) \beta_t(j)}{\sum_{j=1}^N \alpha_t(j) \beta_t(j)} \right] \left[\frac{c_{jk} N(o_t, \mu_{jk}, U_{jk})}{\sum_{m=1}^M c_{jm} N(o_t, \mu_{jm}, U_{jm})} \right]$$

Ezzel a Markov-modellek elméleti alapszintű bemutatásának végéhez értünk.

1.8. 2. Viterbi algoritmus

A legjobb útvonal keresése az a feladat, aminek megoldása során meghatározható, hogy adott eseményvektort a modell milyen állapotosorozata mentén bocsát ki a legnagyobb valószínűséggel. Ezt 'lágý' Viterbi-algoritmusnak (Viterbi, 1967) nevezzük, és annyiban különbözik a Forward algoritmustól, hogy a legvalószínűbb útvonalakat meg kell jegyezni:

1. segédváltozók inicializálása: $\delta_i(i) = \pi_i b_i(o_1)$
 $\psi_1(i) = 0$

2. rekurziós lépés: $\delta_t(j) = \max_{1 \leq i \leq N} [\delta_{t-1}(i) a_{ij}] b_j(o_t)$
 $\psi_t(j) = \arg \max_{1 \leq i \leq N} [\delta_{t-1}(i) a_{ij}]$

3. terminációs lépés: $P^* = \max_{1 \leq i \leq N} [\delta_T(i)]$
 $q_T^* = \arg \max_{1 \leq i \leq N} [\delta_T(i)]$

4. útvonal visszafejtés: $q_i^* = \Psi_{t+1}(q_{t+1}^*)$

1.9. 3. Viterbi adatok tárolása, törlése

A folyamatos felismeréshez szükség van a Viterbi algoritmus végrehajtása során kiszámolt útvonalak időszakos vágására, különben a rendelkezésre álló memória nem lesz elegendő egy percnyi beszéd felismerésére sem. A Beam Search vágási értékét úgy kell megválasztani, hogy a valós időben történő felismerést lehetővé tegye ugyanakkor a lehető legtöbb variációt megvizsgálhassa az algoritmus. Ekkor - a mai számítógépek kapacitása mellett - átlagosan 5000-10000 állapot versenyez egymással. Minden útvonalpont tárolásához 8 bájt szükséges: 4 bájt a valószínűség leírásához, 4 pedig a helyzet jelöléséhez. Ily módon másodpercenként $7500 \cdot 8 \cdot 100$ bájt = 5.7 MB memória töltődik fel információval. Ilyen feltételek mellett az 512 MB memóriakapacitás (melyből az operációs rendszer a felét foglalja) kb. 45 másodperc alatt felemészthető. Ezért van szükség mindenképpen a múltbeli útvonalak elhagyására.

Az adatok eldobása a következő elv szerint zajlik:

Minden egyes Viterbi-algoritmus kiértékeléskor (200ms-onként) egy 10 másodperces szakaszt vizsgálunk, melynek első 20ms hosszú része lesz az eldobandó információ. (10 másodperc nem feltétlenül szükséges, de ebben az esetben már elenyésző a valószínűsége annak, hogy romoljon a felismerés valószínűsége az eldobás miatt.) A szakasz útvonalainak kiértékelése után az útvonalak gyökereit töröljük, de a valószínűség értékeket továbbra is azokból származtatjuk. Így azok hatása a versengő útvonalakra továbbra is megmarad.

A fent leírt módszerrel akkor kerülhető el a memória telítődése, ha folyamatosan a régi helyekre töltődik fel az új információ. Ebben az esetben az aktuális hosszabb-rövidebb útvonalsorok mindig ugyanarra a helyre kerülnek. Ehhez azonban a 10 másodperces szakasz tökéletes lefedése kell, amelynek lényegesen több információ tárolásának lehetőségét kell magában tartania, mint amennyire valóban szükség van. (A Beam Search vágási kritériuma engedélyezi az adatok 'Burst'-ös elhelyezkedését, azaz lehetséges, hogy egy kiválasztott intervallumban az összes, vagy ahhoz közeli számú állapot verseng.) Így $10 \cdot 100 \cdot N \cdot 8 \cdot 3 \cdot 8$ bájtra, azaz ~1800 MB memóriára van szükség. (Ez esetben feltettük, hogy $N=10000$ szó és szavanként átlagosan 8 fonéma verseng az algoritmusban.) Ez a jelenlegi erőforrások mellett sajnos nem lehetséges, így azt a változatot kell választani, amikor a memóriában az adatokat az operációs rendszer által választott véletlen pozíciókban tároljuk. Ekkor megvan az esélye, hogy olyan memóriakiosztás áll elő, amely mellett az új adatsort csak a virtuális memóriát igénybe véve lehet elhelyezni, ez pedig a valós felismerés hamaros leállítását vonja maga után. Az ilyen leállások valószínűsége igen kicsi, mivel azt egyre nagyobb méretű adatsorok tárolása okozhatja, az adatsorok mérete pedig felülről korlátos és azon belül valamilyen normálshoz közeli eloszlást követ. (Így egy új nagyméretű adatsor valószínűleg elfér egy régebbi, már törölt nagyméretű adatsor helyén.) A hiba előfordulása minimalizálható a felismerő hosszú szünetekben történő ismételt inicializálásával, amikor a memóriát felszabadítjuk és a felismerő processzt újraindítjuk.

1.10. 4. Információk szinkronizálása

Ahhoz, hogy az eredményeket folyamatosan meg lehessen jeleníteni annak ellenére, hogy 200ms időközönként egy múltbeli 10 másodpercnyi intervallumot értékel ki a felismerő, szükség van a már kijelzett és a kijelzendő információk egyeztetésére. A karaktorsorozatok esetében a következő alapvető lehetőségek merülnek fel:

Régi sorozat	Új sorozat	Ok	Teendő
abcd	abcd	nincs új adat a 100ms alatt és nem is vettünk adatot t-10s időpontban	nincs
abcd	abcde	új adat jelent meg	y küldése
abcd	bcd	adat veszett el a t-10s időpontban	abcd törlése és bcd újraküldése
abcd	bcde	az előző két eset együttesen	abcd törlése és bcde küldése

Az ilyen esetekben a felismerő nem változtatott a régi, már megjelenített információkon. Ezekon felül a következő speciális lehetőségek merülnek fel:

Régi sorozat	Új sorozat	Ok	Teendő
abcd	abd	Változás történt az új 100ms közben	cd törlése és d küldése
abcd	bd	az kiértékelés változott, valamint a t-10s időpontban adatvesztés is történt	cd törlése d küldése

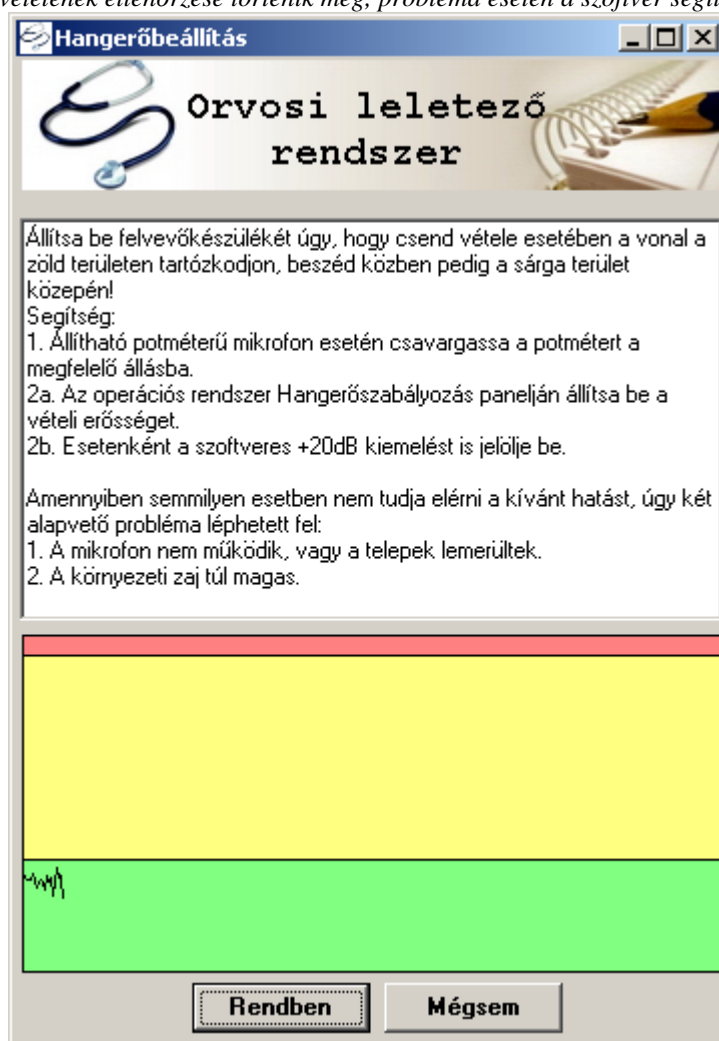
Régi sorozat	Új sorozat	Ok	Teendő
abcd	bde	új információ is megjelent	cd törlése és de küldése

A fent vázolt eseteket a program lekezeli. Elméletileg elképzelhető, hogy a speciális esetek közötti második-harmadik esetben az 'a' információ nem a t-10s időpontban történő természetes adatvesztés miatt tűnt el, hanem a felismerő változtatta meg a döntését. Annak az esélye azonban elenyésző (a teljes tanító- és tesztanyagban nem volt rá példa), hogy 10 másodperccel későbbi információ megváltoztassa a döntést, így ezt az esetet lekezelés nélkül lehet hagyni.

1.11. 5. Vizuális interfész

Az orvosoknak szánt végleges verzióban a felhasználóknak egyéni azonosítójuk van, amely segítségével a nekik megfelelő anyagot lehet kiválasztani.

Ezek után a mikrofon vételének ellenőrzése történik meg, probléma esetén a szoftver segítséget nyújt.



Ha teljesítettük a mikrofonnal kapcsolatos beállításokat, megjelenik a program főablaka, melyen a diktálást el lehet végezni. Az ablak bal oldalán található egy két oszlopból álló hangerőszabályozó. A bal oldali oszlop kék színű kitöltöttséggel jelzi az aktuális hangerőt, míg a jobb oldalival beállíthatjuk a kívánt küszöbértéket. Ennek hatására csak azon hangok fognak a felismerő bemenetére jutni, melyeknek hangereje meghaladja a küszöbértéket. Így zajos környezetben feljebb, csendes környezet esetén pedig lejjebb érdemes a csúszkát mozgatni. (Ha szükséges)

A felső gombsor környezetében található egy színes (zöld, vagy piros) téglalap. A téglalap zöld színe jelzi, hogy a felismerő kapott hanginformációt, míg a piros szín arra utal, hogy a rendszer -kellő hangerő hiányában- nem dolgozik.

A gombok jelentése a következő:

- Lelet indul, Lelet vége: ezzel a gombbal lehet elkezdni, illetve befejezni a lelet diktálását.

- *Mentés: a szövegmezőben lévő szöveget tetszőleges helyre elmenthetjük.*
- *Nyelvi tanítás: ha egy szöveget elmentettünk, akkor lehetőség van a benne lévő nyelvtani elemek megtanítására. Ha ezt a gombot megnyomjuk, akkor a nyelvtani adaptáló program elindul és a szükséges kérdések feltevése/megválaszolása után a rendszert frissíti. (Figyelem! Az új nyelvtan használatához ki kell lépni a diktálási részből és újból kezdeni a mikrofonbeállítástól.)*
- *Vágólapra: a szövegmezőben lévő szöveg a vágólapra kerül, így az a CTRL+C, vagy a Copy/Paste segítségével bármelyik dokumentumba beilleszthető.*

A programból bármikor kiléphetünk azt Alt+F4 billentyűkombináció, vagy a bezárógomb segítségével.

1.12. 6. A programról – fejlesztőknek

A szoftver kódja mindenhol dokumentálva van a kellő mértékben, hogy a Markov-modellek mély szintű megismerése után a programozó bármilyen fejlesztésbe kezdhessen. Kiemelendő azonban, hogy a szükséges matematikai, statisztikai, programozási ismeretek és a szoftver memóriamenedzsmentjének teljes feltérképezése nélkül semmiképpen se kezdjünk bele lényeges módosításokba. Az eredeti szoftver MD5-hash ellenőrző kóddal rendelkezik, minden olyan alkalommal, amikor ellenőrzi a szoftvert, annak forráskódját ellátja ezzel a jelöléssel. Így az illetéktelen, nem egyeztetett változtatások azonnal kiszűrhetők.

2. 4. fejezet – Disztribúció

A szoftver el lett látva telepítőprogrammal, így egy új számítógépre való installálása nem tart tovább néhány kattintásnál. (Fejlesztői munka esetén azonban a forrásokat manuálisan kell másolni.) A telepítéskor néhány ellenőrzést is végez a program, melynek során csak az engedélyezett számítógépekre engedélyezi a telepítést. A telepítéshez szükséges minimális konfiguráció:

- *Pentium, vagy újabb processzor*
- *Windows 98 Se, Windows XP operációs rendszer. (Fejlesztéskor kizárólag XP-n történt a tesztelés, régebbi operációs rendszer esetében a helyes működés nem garantált.)*
- *több, mint 256 MB memória*
- *legalább 800x600-as képernyőfelbontás 16 bites színmélységben.*

Ezen felül a program minden futáskor ellenőrzi a számítógép aktuális sebességét. 1500 MHz alatti frekvencia esetén a szoftver egy üzenet után automatikusan bezár, ugyanis ez a sebesség nem elegendő a futtatáshoz. 1900 MHz alatt a szoftver egy figyelmeztető jelzést küld, mivel ilyen sebességnél elképzelhető, hogy a valós idejű munka érdekében a program az információ egyes részeit eldobja. (Csökkentett mód)

Ajánlott konfiguráció:

- *512 MB vagy több RAM*
- *2GHz-es, vagy gyorsabb processzor*

Disztribúció folyamán a telepítést végző személynek lehetősége van a felhasználók bejegyzésére, de ezt a felhasználók maguk is elvégezhetik az első bejelentkezéskor. Fontos tudnivaló, hogy egy névvel (azonosítóval) az adaptációk miatt csak egyféle munkát lehet végezni. Azaz, ha XY névvel Z munkára (pl. Gastroszkópia) jegyeztük be magunkat, akkor W munkát (pl. Ultrahang) ugyanezzel az azonosítóval nem fogunk tudni végezni. Ilyen esetben a felhasználónak több azonosítóval kell rendelkezni, amelyeket meg tud különböztetni egymástól.

2. sz. Melléklet

A gasztroszkópiai leletező klinikai véleménye, tesztelési tapasztalatai

A projekt indokoltsága

Az endoszkópos munka folyamatában 4 fontos részt különítünk el: beteg felvilágosítás, endoszkópos vizsgálat elvégzése, lelet rögzítés, beteg felvilágosítás. Az endoszkópos vizsgálat tervezett ideje gyomor vizsgálat (gastroszkópia) esetén 15 perc. Ebből ma a vizsgálat maga gyomor tükrözés esetén 7-8 percet vehet, a lelet rögzítés 3-4 percet vesz igénybe. Ez az egész vizsgálat idejét figyelembe véve viszonylag nagy idő.

Naponta egy gastroscopos laborban 13-15 vizsgálatot végzünk el.

Amennyiben a vizsgálat folyamán, azzal párhuzamosan tudnánk rögzíteni a leletet, akkor a látott kép leírását jobban tudnánk rögzíteni, a gastroscopos vizsgálat idejét le lehetne rövidíteni.

Az endoszkópos labor részei és a hangfelismerés alkalmazási lehetőségei

Egy endoszkópos laboratórium a következő egységekből áll: gasztroszkópia amely a nyelőcső, gyomor patkóbél vizsgálatát teszi lehetővé, ERCP(endoszkópos retrográf cholangiopancreatographia,) amely a hasnyálmirigy és az epeutak vizsgálatára alkalmas, Colonoszkópia a vastagbél vizsgálatára, Endoszkópos UH a gyomor és a vastagbél ultrahangos vizsgálatát támogatja.

Az endoszkópos laborban 8-9 orvos dolgozik váltásban, 6-8 nővér segít a betegek vizsgálatában. Leletek kiadását csak orvos végzi. A leletek kiadása a megírás után történik. A leletek írása egy kórházi vagy labor informatikai rendszer keretén belül történik, amely a beteg személyes adatait ill. előző leleteit is tartalmazza.

Az endoszkópos laborban a hangfelismerésnek több alkalmazási lehetősége is van:

Leletírás: mint klasszikus alkalmazás időnyerés, leírások pontos rögzítése céljából.

Műszerek irányítása : az orvos a vizsgálat alatt több műszer párhuzamos irányítását kell hogy megoldja: Az endoszkópon a fény irányítása (4 irány), biopszia vétel, levegő fújás, szívás, képfelvétel nyomtatóra, DVD-re vagy videóra, endoszkópos megvilágítás típusának irányítása.

Ezt vizsgálat közben az orvos aki a vizsgáló műszert tartja, másik kezével irányítja csak akkor képes megoldani, ha az asszisztens átmenetileg átveszi az eszközt.

A gasztroszkópos lelet

A gastroscopos lelet strukturálisan 5 fő részből áll:

Anamnézis, lelet, diagnózis, javaslat, megjegyzés

Az anamnézis részben röviden 1-2 mondatban, szabad szavakkal írjuk le miért került sor a vizsgálatra. Volt-e a beteg már korábban ilyen vizsgálaton, szed-e valamilyen releváns gyógyszert

Lelet struktúrája két fajta lehet: visszafelé haladás során rögzítjük sikerült-e az eszközt a duodenumba vezetni. Ezek után leírjuk a postbulbaris, bulbaris duodenum (patkóbél) elváltozását. A pylorus (gyomor kimenet) formájának, átjárhatóságának leírása után a gyomor 4 részének (antrum, corpus, fornix és fundus) leírása következik. Itt meg kell adnunk felszíni nyálkahártya elváltozásokat, azok helyét, méretét, formáját, a hozzávezető redők elváltozásait. A gyomor leírásánál fontos a perisztaltika (mozgás) jellemzése az esetleges gyomor tartalommal együtt. Az endoszkópos eszköz visszafordításával ún. retroverzióban lehet megvizsgálni vajon a gyomor/nyelőcső átmenet nyitva van-e ? A gyomor nyelőcső átmenetben esetleges

sérv leírása jöhet szóba az átmenet leírásán túl. A nyelőcső hosszanti leírása után még a garat képletek vizsgálata jöhet szóba.

Egy negatív, alap lelet a következőképpen néz ki:

Az eszközt akadálytalanul a duodeumba vezetjük. A megtekintett postbulbaris és bulbaris duodenum ép. A pylorus kerek átjárható. Az antrum, corpus, fornix és fundus eltérés nélkül. A perisztaltika normális. A gyomortartalom kismennyiségű és tiszta. A cardia zárt. A nyelőcső átmenet eltérés nélkül. A nyelőcső és garatképletek épek.

A leletezés másik lehetősége hogy fentről lefelé haladva írjuk le látott képet.

A projekt előzménye

A Semmelweis Egyetem II. Belklinikáján közel 10 éves története van a hangfelismerő rendszerek alkalmazási lehetőségeinek kutatásának. A 90-es évek közepén a Dragon (USA) cég technológiáját alkalmazva kíséreltünk meg gastroscopos leletező rendszert kifejleszteni.

Fejlesztéseinkről akkor az Endoscopy c. folyóiratban számoltunk be.

A hangfelismerés akkori lehetőségeiben külön szavas felismerés volt lehetséges. Folyamatos beszéd nem. A komputerek kapacitása, sebessége is lényegesen kisebb volt mint ma.

A projekt keretében rádió frekvenciás hordozható mikrofont szereztünk be. A projektet a szó alapú felismerés nehézkes alkalmazhatósága miatt 2001-ben abbahagytuk.

2001-ben vettük fel a kapcsolatot Dr. Vicsi Klára professzor asszony csoportjával, akik a Budapesti Műszaki Egyetemen saját algoritmusokat fejlesztettek a magyar beszéd speciális jellemzőinek a figyelembe vételével folyékony szavas beszéd felismerésére.

Ahogy a csoport technikai megoldás javaslatait és első eszközeit láttuk, felhagytunk a saját fejlesztéssel és egy esetleges kooperáció keretében terveztük a projektünk folytatását.

A projekt részei és időszakai

2003/4-ben a projekt benyújtása és elfogadása után az elfogadott projekt terv alapján 4 fázisra terveztük a munkánkat

1. Gastroscopos hangminta felvétel az endoszkópos laborban dolgozó orvosoktól beszéd adatbázis felépítése céljából , szöveges adatbázis felépítése
2. Akusztikai és nyelvi modellek felépítése
3. Tesztelés és javítás
4. Bevezetés és beszámoló

Ad.1 Hangminta felvétel 5 orvosnál történt, akik a leírás után diktálták be az endoszkópos laboratóriumban a leletet. A felvételeknél rádió hullámú mikroszkópot is használtunk, amellyel a vizsgálat közben is rögzítettünk leleteket. Ezzel a módszerrel hangmintát rögzítettünk.

A szöveg adatbázis 2500 leletet tartalmazott , amelyet a labor informatikai rendszerünkől nyertünk ki.

Ad.2 Hangmodell felépítés
Egyéni szómodell felépítés lehetősége

Ad.3 Hangmodellek tesztelése

A beszéd alapú leletezés kivitelezhetőségét 20 lelet bemondása alapján teszteltük. Mind a húsz lelet tartalmazott anamnézis, vélemény, diagnózis, javaslat, megjegyzés részeket.

A tesztelés folyamán mértük a leletezés sebességét az íráshoz képest , a leletezés közben felmerülő hibákat, azok típusát és helyét a leletben. Hibaként szócserelést, betű hibát különböztettünk meg.

A tesztelést egyéni szómodell nélkül és az egyéni szómodellel is elvégeztük.

A 20 lelet átlagban 70 szót tartalmazott, átlagos bemondási hosszuk 40mp volt. Az átlagos Szótévesztés induláskor 25% volt. A felismerési idő pedig a bemondási idő 1,5-2-szerese. Tesztelés, javítás után a szótévesztés 17%-ra csökkent, a felismerési idő pedig kisebb, vagy egyenlő lett a bemondás idejével, vagyis a felismerő valós időben működik.

Ezek helye 64%-ban az anamnézis, 20%-ban a vélemény, 6%-ban a diagnózis, 6%-ban a javaslat, 4%-ban a megjegyzés volt.

Ezeket az eredményeket beszélő független felismerés esetében kaptuk.

A fejlesztés során lehetőség nyílt az egyéni szófordulatokhoz való alkalmazkodásra, vagyis a használat során a felismerő rendszer a használó saját stílusához alkalmazkodik. Ebben az esetben az egyéni szómodellek alkalmazásával a 20 lelet tekintetében a következő eredményeket kaptuk.

A 20 lelet átlagban 70 szót tartalmazott. Ezeket 40 mp-nyi idő alatt rögzítettük. felismerési idő pedig kisebb, vagy egyenlő volt a bemondás idejével. Az átlagos szóhiba szám pedig 6%-ra csökkent. Ezek helye a 68%-ban az anamnézis, 20 %-ban a vélemény, 4 %-ban a diagnózis, 4%-ban a javaslat, 6%-ban a megjegyzés volt.

A projekt tapasztalatai

1. Felismerési pontosság: A felismerési pontosság eléri azt a szintet a „vélemény”, a „diagnózis”, „javaslat” és „megjegyzés” részek tekintetében amivel rutinben is tudjuk használni a rendszert. Az anamnézis tekintetében még nem elfogadható a szó és betű pontosság általános szótárak használata esetén. Az anamnézis rész tekintetében a nyelvi adaptáció, felhasználó specifikus szótárak és hageminta adatbázisok alkalmazásával lehet elérni a rutin felhasználás szintjét.

2. Sebesség: A diktálás sebessége eléri ill. túlhaladja egy átlagos gépiró sebességét.

3. Javítási, tanítási lehetőségek: A tanulási opció a programban lehetővé teszi hogy személyre szólóan fejlesszünk ki és alkalmazzunk beszéd felismerési algoritmusokat. Ezzel a gyakorlati alkalmazásokhoz szükséges beszéd felismerési biztonságot el lehet érni.

Alkalmazási lehetőségek

A hangfelismerő rendszert rádió mikrofonos és telepített mikrofonos üzemmódban lehetséges használni. A leletek a billentyűzet driveren keresztül, mintha billentyűzeti információ jelenik meg a képernyőn. Így lehetséges billentyűzet alternatívájaként használni. Klinikánkon az E-MEDSOL rendszer érhető el.

Hangfelismerés integrálása a laborinformatikai szoftverbe az EMEDSOL környezetbe

Az alkalmazás helye az endoszkópos leletezési folyamatban a vizsgálat ideje ill. az utána következő percek lehetnek. A beteg személyes adatait az E-MEDSOL rendszerben behívva, a leletezés menüpontba lépve a hangfelismerést lehetséges aktiválni. A leletet bediktálva lehetséges javítani billentyűzettel még mentés előtt. A leletezés végén visszatérve az E-MEDSOL parancsmenü rendszerbe lehetséges a leletet menteni.

A projekt eredményeinek ismertetése

Az hangfelismerésen alapuló endoszkópos leletezés lehetőségeiről, javaslatairól, eddigi tapasztalatainkról beszámolunk a Magyar Gasztroenterológiai Társaság konferenciáján 2007-ben ill. Digestive Disease Week konferencián az Egyesült Államokban ill. a United European Gastroenterology Week rendezvényén, 2007 évben.

Budapest, 2006. október 30.

Prof. Dr. Molnár Béla
tudományos főmunkatárs
Semmelweis Egyetem II. Belklinika

3. sz. melléklet

A hasi és kismedencei ultrahang vizsgálat leletező klinikai véleménye, tesztelési tapasztalatai

A Semmelweis Orvostudományi Egyetemen nemrégén került bevezetésre a MedSolution (E-MEDSOL) kórházi számítógépes információs rendszer, amely a betegek kezelésének részleteit, így kórrajzokat, az elvégzett vizsgálatok, beavatkozások, esetleges műtétek leírását, zárójelentéseket, stb. tárol digitális formában az egyes betegekről egyfajta „digitális kartonozó” formájában. Ennek megfelelően a leletek ma már számítógépes szövegszerkesztővel készülnek, a radiológiai vizsgálatok eredményeit a vizsgálatot végző szakorvos lediktálja asszisztensének, aki azt beviszi a számítógépes rendszerbe. Ez a bevitel történhet azonnal a vizsgálat után, de előfordul, hogy a szakorvos diktafonra diktálja fel a leletet, amelyet később az asszisztensek lehallgatnak és szövegesen begépelnek. Ez utóbbi feladathoz szakszerűen kiépített megfelelő munkaállomások állnak rendelkezésre. Így történik ez a radiológiai, ultrahangos vizsgálatok eredményeivel is.

A számítógépes technológia, illetve a számítógépes beszédfeldolgozás fejlődésével lehetővé vált, hogy az emberi beszédet felismerő és azt szöveges formába átalakító szoftvereket fejlesszenek ki. Ehhez a lelet szövegét digitális formában rögzíteni kell, ami a ma már széles körben elterjedt PC-khez kapcsolódóan egy jó minőségű hangkártya és egy megfelelő jelátalakító, azaz mikrofon használatát igényli. Ezek az eszközök ma már könnyen beszerezhetőek, és árukat tekintve is megfizethetőek, a számítógéphez való csatlakoztatásukat pedig minimális számítógépes ismeretekkel rendelkező személy is könnyedén el tudja végezni. Ezután a vizsgálatokat elvégző szakorvosnak semmi más dolga nincs, mint elindítja a beszédfelismerő alkalmazást, és lediktálja a leletet úgy, ahogyan eddig is tette. A program ablakából a bediktált lelet könnyedén exportálható és a MedSolution-ba beilleszthető.

A számítógépes beszédfelismerő használatával azonban lényegesen leegyszerűsíthető a leletezés arányaiban legnagyobb ráfordításigényű művelete, a lelet begépelése. A megfelelően beállított, betanított rendszer ugyanis az emberi beszéd felismerésével azzal egyidejűleg folyamatosan írja át a diktált leletet szöveges formába. Jóllehet a beszédfelismerő rendszerek bizonyos hibaszázalékkal üzemelnek, azaz esetenként előfordulhat, hogy a diktált lelet egy-egy szavát, kifejezését „félreértik”, ekkor a felismerő téveszt. Ha a tévesztések aránya kellően alacsonyan tartható, azaz egy-egy leletben legfeljebb néhány szó felismerése hibás, akkor a rendszer kitűnően használható. A hibásan lejegyzett szavakat ezután kézzel, begépeléssel javítani kell, arányaiban azonban jóval kevesebb gépelés szükséges, mintha az egész leletet a hagyományos úton vinnék be. A javítást végezheti maga az orvos is, közvetlenül a diktálást követően, hiszen aláírás előtt mindenképpen átolvassa az elkészült leletet. Mindemellett természetesen lehetőség van arra is, hogy az asszisztens javítsa ki az esetleges hibákat, ehhez esetleg a lediktált leletet magát is visszahallgathatja.

A beszédfelismerő rendszerek használata előtt a felismerőt adaptálni szükséges az adott felhasználási körülményekhez, illetve felhasználóhoz kell illeszteni. Ennek egyik része az ún. akusztikai tanítás. Ekkor a rendszer nem tesz mást, mint megtanulja az adott felhasználó beszédének akusztikai jellegzetességeit. Alapvető követelmény, hogy a rendszer használata előtt, amikor az adott felhasználót létrehozuk, adjuk meg a felhasználó nemét is, a férfi és női beszéd között ugyanis nagy különbség van, így ennek megadásával megkönnyítjük a felismerő munkáját. Szintén ügyelnünk kell arra, hogy a program használata során a mikrofonról

felvett jel elég erős legyen az esetleges más, környezeti zajokhoz képest. Ezt a program indításakor minden esetben megjelenő ablakban végezhetjük el, a mikrofonról vett jel erősségét egy görbe szemlélteti, amelynek beszéd esetén a sárga, csend esetén a zöld területű tartományban kell futnia.

Az adaptálás második, kissé bonyolultabb, de idővel könnyen elsajátítható része a nyelvi tanítás. Ekkor a beszédfelismerő rátanul az adott felhasználó – azaz szakorvos – jellemző szófordulataira, beszédstílusára, ami nagymértékben javítja és gyorsítja a felismerést. A módszer hasonlóan működik ahhoz, mint amikor az asszisztens, aki már hosszabb ideje dolgozik az adott orvos mellett, sok esetben gyakorlatilag előre tudja, hogy a szakorvos hogy fog egy-egy adott mondatot megfogalmazni, azaz akár előre tudja, mit fog mondani. Ha a felismerő is megtanulja ezt, jóval kevesebb hibát fog véteni, ehhez azonban el kell telnie egy bizonyos időnek, amíg elegendően nagy számú leletet „hall” az adott orvostól. Ehhez a rendszert tanítani kell, ezt jelenti a nyelvi tanítás. Ennek során a felhasználó már korábban lediktált, kijavított és elmentett szöveges leleteit alapul véve a rendszer félautomatikusan frissíti tudását. A tanítás azért félautomatikus, mert a felhasználónak egyrészt kijavított leleteket kell elmentenie, hogy azokat a rendszer helyesen tanulhassa meg, másrészt, ha az újonnan bementett leletek ismeretlen szót tartalmaznak, annak kiejtését meg kell adnunk, hiszen a rendszer azt nem tudja kitalálni. Ezen túlmenően azonban minden frissítés és adaptáció (ún. nyelvi adaptáció) automatikusan zajlik, amelynek sikeres lezárulásáról a felhasználó üzenetet kap. Ezt követően a leletező rendszerbe történő újabb bejelentkezés után már az új, frissített nyelvi beállítások lesznek aktívak.

A leletező beszédfelismerőt jelenleg két belgyógyászati vizsgálati területen lehet használni, amelyek közül a felhasználó felvételekor kötelező jelleggel választani kell. E két téma terület a gasztroenterológia, ezen belül is a gyomor- és vékonybél tükrözés, illetve a radiológia, amely jelenleg a hasi és kismedencei ultrahang vizsgálatok leleteinek elkészítésére alkalmas. Igény esetén lehetséges további téma területek bevonása is, így az ultrahangos vizsgálatokat ki lehet terjeszteni (pl. kolor Doppler vizsgálatok, nyak, pajzsmirigy ultrahangos vizsgálata, echocardiographia, stb.), illetve egyéb radiológiai vizsgálatok leletezése is minden további nélkül megoldható elviekben a rendszerrel (Röntgen, CT, MRI vizsgálatok). Hasonlóképpen a gasztroenterológiai területen a vastagbél tükrözése (colonoscopia), az endoszkópos ultrahang (EUS), és az eperendszer és pancreas képalkotó vizsgálatai (ERCP) kerülhetnek szóba az első alkalmazási terület-bővítési körben. E bővítéshez a felismerő programot nem kell változtatni, azonban az adott területre adaptálni kell, ami a programfejlesztők és az adott területen tevékenykedő szakorvosok együttes munkáját feltételezi.

A rendszer használata során szerzett tapasztalataink összességében pozitívak, a beszédfelismerést leletezéshez 2 fő szakorvos tesztelte a radiológiai osztályon hasi és kismedencei ultrahang vizsgálatok leleteinek elkészítéséhez. Emellett rendszeres konzultáció történt a programfejlesztőkkel is, az alábbiakban néhány közös észrevételünket, megjegyzésünket szeretnénk közreadni értékelésképpen. A rendszer előnyei véleményünk szerint:

a gyors, valós idejű működés, a lelet hamarabb készül el, mint a hagyományos begépelés esetén;

a nyelvi tanítással a rendszer teljesítménye észrevehetően javul;

az „unalmas”, sokszor ismételt kifejezések felismerése rendkívül pontos;

a leletekben nincsenek elgépelési hibák, nem szerepelnek benne olyan rövidítések (pl. belső használatú házi rövidítések), amelyek az érthetőséget nehezítenék;

a rendszer a bonyolultabb kifejezéseket is helyesen írja le;

a latin kifejezéseket a beszédfelismerő érti és egységesen, helyesen írja le;

a diktálás gyorsítása érdekében bizonyos rövidítéseket a rendszer felismer (pl. st. p.; l.s.; l.d.; l.u.; vs.; stb.)

a lelet diktálása bármikor felfüggeszthető (pl. ha a szakorvos telefont kap, vagy máshoz beszél), majd ugyanonnan folytatható;

a lelet javítása korszerű, kényelmes szövegszerkesztő ablakkal történik (szemben a MedSolution jelenleg linux szerveren futó, ezért az átlagos felhasználó számára bonyolult szerkesztési parancsaival);

a lelet könnyedén exportálható a MedSolution információs rendszerbe.

A rendszer használatánál néhány dolog különös figyelmet érdemel:

a számítógép „audio és hangeszközök” beállításait rendkívül gondosan, a programfejlesztők utasításainak maradéktalan betartásával kell elvégezni, különben a rendszer gyengébb teljesítménnyel működik;

a leletek mentésekor csak teljesen kijavított lelet menthető, mert ellenkező esetben az elvégzendő nyelvi tanítás során bevisszük a rendszerbe a hibát, ami ahelyett, hogy javítana, még ronthat is a működésen. Ilyen esetben egyébként lehetőség van a tanítás előtti nyelvi beállítások visszaállítására, de ekkor minden addigi nyelvi tanítás eredménye elveszik.

A betegek nevének felismerése nem várható el a rendszertől, hiszen rendkívül nagy az egyes vezetéknevek száma, ráadásul hangzás alapján nem tudnánk megkülönböztetni az „y”-nal írandó neveket az „i”-vel írandóktól, a „cs”, „ts”, stb. variációk elkülönítése is megoldhatatlan. Ezért a beteg nevét és egyéb személyes adatait továbbra is kézzel kell begépelni a MedSolution-ba.

Speciális, nem tipikus vizsgálat lefolyása, illetve az ilyen vizsgálatok során látottak dokumentálása megoldható a felismerővel, de ilyenkor nagyobb a rendszer hibaaránya, és előfordulhat, hogy akár egész mondatok felismerése hiúsul meg. Ekkor kis türelemre van szükség a kézi javításhoz, amely után azonban a nyelvi tanítással biztosítható, hogy a rendszer a jövőben jobb eredményt adjon az adott típusú leleteket illetően (ilyen eset lehet pl. IBD ultrahangos diagnosztikája, ill. bizonyos hasi vagy kismedencei térfoglaló folyamatok körülírása, mivel ilyen típusú leletek a rendszer eredeti nyelvi tanítóanyagában relatíve kevesebbszer fordultak elő).

A leletező program csak olyan számítógépen futtatható, amely legalább 1,6 GHz-es processzorral és 512 MB RAM-mal rendelkezik (a processzorteljesítmény általában elégséges, a memória szükség esetén bővíthető).

Javaslatunk a rendszer továbbfejlesztését illetően:

mindenképpen szükségesnek tartjuk, hogy ne csak a hasi és kismedencei ultrahang leletezését lehessen a felismerővel megoldani, mivel a gyakorlatban sokszor egyszerre végeznek hasi és nyaki (pajzsmirigy) vizsgálatot, ill. pl. Doppler vizsgálatot;

a rendszer a kimeneten a számokat is latin betűs szöveges átiratban jeleníti meg, ami kissé nehezíti a lelet olvasását;

kívánatos lenne a leletek sortöréssel való tagolása az egyes szervek, szervcsoportok szerint a jó áttekinthetőség érdekében;

a felhasználói interfész áttekinthetősége jó, de az egyes funkciók csoportosítása hasznos lehet a kevésbé tapasztalt felhasználók számára;

előnyös lehet a felismerő kliens-szerver alapú működtetése legalább az akusztikai és nyelvi modulok megosztásával, mivel az orvosok több vizsgálóhelységet használnak felváltva (a relatíve nagy memória és CPU felhasználási igény miatt a szerveren való futtatás háttérbe szorul);

néhány esetben a felhasználót megzavarhatja, hogy a tévesztések bizonyos esetben kumulá-

lódnak, a felismerő „nehezen talál vissza, ha egyszer eltévedt”; az alkalmazás kezdeti időszakában (első 1-2 hét) technikai segítség a szakorvosok számára mindenképpen szükséges.

Végezetül néhány nyers adat a rendszer működésével kapcsolatosan az alábbi táblázatban olvasható. A beszédfelismerő rendszerek működését a helyesen felismert szavak arányával szokás jellemezni (hatékonyság). A helyesen felismert szavak számát megkapjuk, ha a leletben előforduló összes szavak számából kivonjuk azokat a szavakat a számát, amelyeket a rendszer eltévesztett (helyette más szót írt ki) vagy kihagyott. A pontosság mérőszámában figyelembe vesszük azt is, ha esetleg a felismerő olyan szót szűr be a szövegbe, amit nem diktáltunk. Ha ezen szavak számát levonjuk a helyesen felismert szavak számából, és osztunk az összes szó számával, megkapjuk a pontossági arányt. Mindez a tesztelés során bemondott mintegy 50 leletből véletlenszerűen kiválasztott 5 leletre az alábbiak szerint alakul:

1. táblázat: A szakorvosok által bemondott leletek felismerési hatásfoka néhány leletre

Lelet azonosító	Összes szó a leletben	Helyesen felismert	Tévesztett	Kihagyott	Beszúrt	Hatékonyság (Corr, %)	Pontosság (Acc, %)
3114550	56	53	2	1	0	94,6	94,6
3125451	98	92	4	2	2	93,6	91,8
3126000	90	86	3	1	3	95,6	92,2
3128910	54	53	1	0	1	98,1	96,3
3128921	61	38	21	2	5	62,3	54,1

A táblázat utolsó sorában látható rosszabb eredmény oka véleményünk szerint az, hogy a lelet szövegezése jelentősen elütött a szokványostól. Nyelvi tanítás után ugyanis az adott lelet ismételt bemondásával kapott eredmény 93,4% hatékonyság, 93,4% pontosság.

A helyszínen, tehát a kórházban használat közbeni tesztelés előtt a Budapesti Műszaki és Gazdaságtudományi Egyetem Beszédkusztikai Kutatólaboratóriumában előzetesen már tesztelték a hasi és kismedencei ultrahangos leletező beszédfelismerőt szintén a SOTE II. Belgyógyászati Klinikájáról származó 15 darab, eredeti lelet bemondásával. Ezek eredménye az alábbi táblázatokban látható. A 2. táblázat férfi beszélők, míg a 3. táblázat női beszélők hanganyagán végzett teszteredményeket mutatja nyelvi tanítás előtt, azaz a rendszer alapállapotában, illetve nyelvi tanítás után. Az eredmények láthatóan javulnak a nyelvi tanítás hatására, különösen az olyan leletek esetében, amelyek felismerése eredetileg az átlagosnál rosszabb arányú volt. Ennek oka, hogy ezek a leletek a tipikus leletszövegeztől eltérőek, vagy a rendszer számára ismeretlen szavakat tartalmaznak. A rendszer a nyelvi tanítás után már ismeri ezeket a számára új elemeket is.

2. táblázat: Férfi beszélők által bemondott leletek felismerési hatásfoka

Lelet azonosító	Nyelvi tanítás előtt		Nyelvi tanítás után	
	Hatékonyság (Corr, %)	Pontosság (Acc, %)	Hatékonyság (Corr, %)	Pontosság (Acc, %)
Teszt_01	91,7	85,0	95,0	91,7
Teszt_02	92,8	84,1	95,7	91,3
Teszt_03	76,8	73,2	91,1	91,1
Teszt_04	93,6	91,5	95,7	93,6
Teszt_05	94,9	91,5	94,9	91,5
Teszt_06	95,1	92,7	95,1	92,7
Teszt_07	87,5	75,0	98,2	94,6
Teszt_08	81,1	73,0	91,9	86,5
Teszt_09	96,0	94,0	96,0	94,0
Teszt_10	91,1	86,7	91,1	86,7
Teszt_11	94,3	91,4	94,3	91,4
Teszt_12	91,9	87,1	91,9	88,7
Teszt_13	97,2	94,4	97,2	94,4
Teszt_14	92,5	82,5	97,5	92,5
Teszt_15	90,5	84,1	98,4	93,7
Összesítés	91,7	85,0	95,0	91,8

2. táblázat: Női beszélők által bementett leletek felismerési hatásfoka

Lelet azonosító	Nyelvi tanítás előtt		Nyelvi tanítás után	
	Hatékonyság (Corr, %)	Pontosság (Acc, %)	Hatékonyság (Corr, %)	Pontosság (Acc, %)
Teszt_01	91,7	86,7	95,0	90,0
Teszt_02	88,4	82,6	92,8	89,9
Teszt_03	78,6	71,4	98,2	94,6
Teszt_04	91,5	91,5	93,6	95,7
Teszt_05	93,2	93,2	91,5	91,5
Teszt_06	97,6	95,1	97,6	95,1
Teszt_07	91,1	83,9	96,4	94,6
Teszt_08	75,7	64,9	94,6	91,9
Teszt_09	98,0	94,0	98,0	94,0
Teszt_10	88,9	84,4	88,9	84,4
Teszt_11	100,0	98,6	100,0	98,6
Teszt_12	96,8	91,9	96,8	93,5
Teszt_13	95,8	93,0	95,8	93,0
Teszt_14	95,0	85,0	97,5	92,5
Teszt_15	88,9	79,4	95,2	92,1
Összesítés	91,8	86,9	95,5	92,9

Összességében elmondható, hogy a rendszer valós felhasználási körülmények között is képes a fenti paraméterekkel jellemezhető működésre.

Budapest, 2006. okt.30.

Dr. Csöregh Éva
A Radiológiai Osztály vezetője
Semmelweis Egyetem II. Belklinika