

Magyar nyelvű, kísérleti e-mail diktáló rendszer

Tarján Balázs¹, Nagy Tímea¹, Mihajlik Péter^{1,2}, Fegyó Tibor^{1,3}

¹ Budapesti Műszaki és Gazdaságtudományi Egyetem,
Távközlési és Médiainformatikai Tanszék
{tarjanb, nagy, mihajlik, fegyó}@tmit.bme.hu

² THINKTech Kutatási Központ Nonprofit Kft.

³ AITIA International Zrt.

Kivonat: Bár a közelmúltban a szélesebb közönség számára is hozzáférhetővé váltak magyar nyelvű diktálórendszerek, használatukhoz állandó internetkapcsolat szükséges, nem teszik ki az írásjeleket és a kis-nagy kezdőbetűk használata sem követi a helyesírási szabályokat. Cikkünkben beszámolunk egy olyan diktálórendszer fejlesztéséről, mely akár a felhasználó eszközén (pl. laptop) futva, egyes írásjelek automatikus elhelyezése mellett képes számok, emotikonok, nagybetűs szavak és rövidítések felismerésére is, így drasztikus mértékben csökkentheti a bediktált szöveg utólagos gondozására fordítandó időt. Ékezetesítő eljárás használatával és a felismerő modellek személyre szabásával 26%-os szóhibarányt értünk el nagyszótáras, e-mail diktálási feladaton. Kísérleti rendszerünkben megvizsgáltuk az egyes írásjelek automatikus elhelyezésének lehetőségeit is. Eddigi eredményeink azt mutatják, hogy csak a „vessző” kiváltására kapható megfelelően pontos előrejelzés a nyelvi modell alapján.

1 Bevezetés

Régi vágyunk, hogy magyar nyelven, viszonylag kötetlen témakörben diktálhassuk elektronikus leveleinket. Noha a közelmúltban a szélesebb közönség számára is megjelentek ilyen alkalmazások (Nuance, Google magyar nyelvű diktálórendszerek okostelefonokra), hamar szembesülniük kellett a felhasználóknak e rendszerek korlátaival. Ilyen például, hogy ezek használatához állandó internetkapcsolat szükséges, hogy a felismerési hibák kisebb-nagyobb százalékban elkerülhetetlenek, a javításuk nehézkes, továbbá nem teszik ki az írásjeleket, és a kis-nagy kezdőbetűk használata sem követi a helyesírási szabályokat. Ráadásul mindkét rendszer távoli szervereken futtatja a felismerést, mely adatvédelmi problémákat is felvethet az arra érzékeny felhasználóknál.

Cikkünkben beszámolunk egy olyan **magyar nyelvű diktálórendszer** fejlesztéséről, mely akár a felhasználó eszközén (pl. laptop) futva, egyes írásjelek automatikus elhelyezése mellett képes számok, emotikonok, nagybetűs szavak és rövidítések felismerésére is, így drasztikus mértékben csökkentheti a bediktált szöveg utólagos gondozására fordítandó időt. A pontosság növelése érdekében egy ékezetesítő eljárást is bevetettünk a tanítószöveg hibáinak javítására illetve személyre szabott felismerő modellekkel is végzünk kísérleteket. Magyar nyelvű kvázi kötetlen diktálásról igen kevés korábbi publikáció született, legjelesebb irodalomnak az [1] tekinthető. Az itt

ismertetett felismerő nagyszótáras, morfoszintaktikai szabályokkal kiegészített, morfémaalapú nyelvi modellen alapult. Fontos megjegyezni ugyanakkor, hogy e korai rendszer gyakorlati hasznát erősen korlátozta, hogy nyelvi modelljét hírlapok szövegén tanították, valamint hogy a diktálást segítő lexikai elemek sem képezték a rendszer részét. Magyar nyelvű diktálási eredményeket emellett még [2]-ben találunk, mely egy kórházi leletező rendszert mutat be. Bár folyamatos diktálásra itt is van lehetőség, azonban csupán közepes szótárméretű, szűk témájú és kis perplexitású felismerési feladaton. Ezzel szemben jelenlegi kísérleteink célja egy, a gyakorlatban is jól használható diktálóalkalmazás létrehozása volt.

2 A kezdeti nagyszótáras e-mail felismerő

Ebben a fejezetben az e-mail diktáló rendszerünk alapjául szolgáló kezdeti nagyszótáras, folyamatos beszéd felismerőt mutatjuk be. Először kitérünk a tanítóadatok begyűjtésével és feldolgozásával kapcsolatos kérdésekre, majd bemutatjuk a felismerő rendszerben használt modellek tanítási lépéseit. A fejezetet a kezdeti eredmények ismertetésével zárjuk.

2.1 Tanítóadatok gyűjtése és előfeldolgozása

Kísérleti e-mail diktáló rendszerünk tanításához olyan szöveges adatbázist kerestünk, mely elegendően nagy egy gépi beszéd felismerő nyelvi modelljének a betanításához, azonban nem tartalmaz bizalmas jellegű, személyes információkat. Ezért esett a választásunk a tanszéki laborcsoportunk belső levelezésére. További előny, hogy a betanított rendszert laborunk tagjai akár a mindennapok során is tesztelhetik, így hamarabb derülhetnek ki az esetleges hibák, és merülhetnek fel továbbfejlesztéssel kapcsolatos ötletek.

Az adatgyűjtés első lépésében a labor minden tagjától begyűjtöttük a leveleket a tanszék alapértelmezett levelezőkliensének tárolási formátumában. Ez a formátum tartalmazza a feladó, címzett, tárgy stb. mezők adatait is, melyet egyelőre a kísérleti rendszerünkben nem vettük figyelembe. A kezdeti rendszer tanításához kivettünk minden írásjelet az e-mailekből. Annak érdekében, hogy meg tudjuk jeleníteni a mondaton belüli nagybetűs szavakat, a szokásos kisbetűsítés helyett egy speciális normalizálást alkalmaztunk [3]. Minden nagybetűs szóalakot eredeti formájában hagytuk, mely alól egyedül a mondatkezdő szavak képeztek kivételt. A mondatok kezdőszavait csak akkor hagytuk meg nagybetűsnek, ha a Hunmorph [4] morfológiai elemző kizárólag ebben az alakban fogadta el őket. A [3]-ben bemutatott módszert követve a számok és a kiejtési kivételszótárban feloldott rövidítések, betűszavak felismerése is lehetővé vált. Minta a kezdeti rendszer tanítószövegének egy sorára:

```
„a Redmine-on keresetem a VOXerver dokumentációját de  
végül nem találtam meg”
```

2.2 Tanítás és dekódolás

A kezdeti felismerő nyelvi modelljének tanításához egy összesen 4 millió szót tartalmazó e-mail korpuszt használtunk fel. A nyelvi modellek – mint minden további kísérleteinkben szereplő modell – módosított Kneser-Ney simítás [5] használatával készültek az SRI Language Modeling Toolkit (**SRILM**) [6] segítségével. A létrehozott 3-gram, szóalapú modellekben entrópiaalapú metszést egyetlen esetben sem alkalmaztunk.

Az e-mail diktálási feladathoz szorosan illeszkedő hanganyag előzetesen nem állt rendelkezésünkre, így egy, a feladattól független akusztikus modellt kellett használnunk a kezdeti rendszerben. A Egri Katolikus Rádió (EKR) beszélgetéseiből válogattott, összesen 43 óra hanganyagot tanított, környezetfüggő akusztikus modell a Hidden Markov Model Toolkit [7] eszközeinek segítségével készült, és összesen 6121 egyenként 13 Gauss-függvényből álló állapotot tartalmaz.

A 16 kHz-en mintavételezett felvételek lényegkiemeléséhez 39 dimenziós, delta és delta-delta értékkel kiegészített mel-frekvenciás kepsztrális komponenseken alapuló jellemzővektorokat hoztunk létre, és ún. vak csatornaki egyenlítő eljárást [8] is alkalmaztunk. A súlyozott véges állapotú átalakítókra (**WFST** – Weighted Finite State Transducer) [9] épülő felismerő hálózatok generálását és optimalizálását az Mtool keretrendszer programjaival végeztük, míg a tesztelés során alkalmazott egyutas mintaillesztéshez a VOXserver [3] nevű WFST dekódert használtuk. A felismerő rendszerek teljesítményének értékeléséhez szóhibarányt (**WER** – Word Error Rate) számoltunk.

2.3 Kezdeti kísérleti eredmények

A teszteléshez összesen 21 perc felolvasott e-mailt használtunk. A felolvasott levelek mind egyetlen feladótól származnak. Ettől a feladótól egyetlen levelet sem tartalmaz a kezdeti rendszer tanítószövege. A kiértékelési eredményeket az **1. táblázatban** foglaltuk össze. A táblázatban található OOV (Out of Vocabulary) arány rövidítés a szótáron kívüli szavak tesztszövegben mutatott arányára utal.

1. táblázat: A kezdeti felismerő kiértékelési eredményei.

	Szótárméret [ezer szó]	OOV [%]	Perplexitás [-]	WER [%]
Kezdeti rendszer	263	5,0	585	38,9

3 Az e-mail felismerő továbbfejlesztése

Cikkünk harmadik fejezetében a kezdeti e-mail felismerő továbbfejlesztésével kapcsolatos lehetőségeket vizsgáljuk meg és értékeljük ki. Célunk az, hogy a diktálást segítő funkciókat egy olyan rendszerbe tudjuk beépíteni, mely jó kompromisszumot képvisel a felismerési pontosság és a komplexitás között.

3.1 A tanítószöveg ékezetesítése

A magyar abc számos ékezetes betűt tartalmaz, melyeket sajnos a nem vagy nem helyesen lokalizált alkalmazásokban nem tudunk bevinni. Másrészt sok felhasználó – így kollégáink közül is többen – a gyors gépelés érdekében az ékezetes betűket ékezet nélküli megfelelőjükkel helyettesíti. Az esetek döntő többségében ez az érthetőséget nem befolyásolja, sőt legtöbbször észre sem vesszük, ha ékezetek nélküli szöveget olvasunk. A felismerő rendszer azonban nem rendelkezik valódi nyelvi intelligenciával, így nyelvi modelljében nem tudja megfeleltetni egymásnak egy szó ékezetes és ékezet nélküli alakját, melynek következtében ugyanazon szókapcsolatot több különböző alakban is modellezzük. Ez rontja a statisztikai becslés pontosságát.

Megoldásként a tanítószöveg **ékezetesítése** mellett döntöttünk. Az ékezet nélküli szóalakok ékezetes változatának megkereséséhez egy speciális szótárat alkalmaztunk, melyet tanszéki kollégáink bocsátottak rendelkezésünkre [10]. Ez a szótár a leggyakoribb ékezetes párjával rendeli össze az ékezet nélküli szóalakokat. Helyzetünket nehezítette, hogy ékezetes és ékezet nélküli tanítószöveg vegyesen állt rendelkezésünkre, így a mindkét alakban értelmes szavakat valahogyan kezelniünk kellett. Kísérleti rendszerünkben azt az egyszerű megoldást követtük, hogy minden ékezet nélküli szóalakat ékezetesítettünk, ha szerepelt a szótárban. Az ékezetesített tanítószöveggel kapott eredményeket a **2. táblázatban** foglaltuk össze. Mint látható, a szótárméret csökkent, hála a kétféle formában létező szóalakok kiszűrésének. Egyedül az OOV arány romlott feltehetően a hibásan ékezetesített szavak miatt, azonban ezt a mért perplexitáscsökkenés kompenzálja, így összességében 2%-os relatív hibacsökkenést sikerült elérnünk.

2. táblázat: Az ékezetesített felismerő kiértékelési eredményei.

	Szótárméret [ezer szó]	OOV [%]	Perplexitás [-]	WER [%]
Ékezetesített rendszer	244	5,4	532	38,1

3.2 A rendszer személyre szabása

A hatékony diktálórendszerek használatba vételét mindig egy tanítási vagy adatgyűjtési feladat előzi meg, ezért úgy döntöttünk, hogy mi is felhasználunk beszélőspecifikus adatokat a rendszerünk optimalizálásához. Első lépésben a diktálórendszer nyelvi modelljét egészítettük ki a tesztanyaghoz tartozó feladó korábbi leveleivel. Ezt az összesen 83 ezer szót tartalmazó tanítószöveget nyelvimodell-interpolációs technika segítségével egyesítettük az ékezetesített kezdeti rendszer modelljével. Az interpolált nyelvi modellek készítéséhez és optimalizálásához az SRILM beépített lineáris interpolációs és perplexitászámító eljárásait használtuk. Az új nyelvi modellel kapott eredményeket a **3. táblázatban** mutatjuk be.

A szöveges adatok mellett az adott beszélőtől származó hanganyagok is felhasználhatóak a rendszer személyre szabása során. A kézi munka minimalizálása érdekében a rögzített tesztanyagon **felügyelet nélküli adaptációt** hajtottunk végre. Az adaptált akusztikus modellel végzett teszt eredményét szintén a 3. táblázat tartalmazza.

Mint az a táblázatból is kiolvasható, a nyelvi modell adaptációval az ékezetesített rendszerhez képest 3%-os relatív szóhiba-arány csökkenés érhető el. Ezen felül azonban további 30%-os javulást mértünk az akusztikus modell adaptálásával. Ez alapján elmondható, hogy a kezdeti nyelvi modell távolról sem állt olyan messze az optimálistól, mint a kiindulás EKR akusztikus modell, mely teljes mértékben a feladattól független adatokon került betanításra.

3. táblázat: A személyre szabott felismerő kiértékelési eredményei.

	Szótárméret [ezer szó]	OOV [%]	Perplexitás [-]	WER [%]
Nyelvimodell- adaptált rendszer	246	5,0	501	37,0
+Akusztikusmodell- adaptáció				26,0

4 Kiegészítő funkciók a diktáláshoz

A korábban fejlesztett felismerőrendszereinkben a beszédet mint szótári szavak sorozatát modelleztük. A közelmúltban azonban eredményesen teszteltünk egy újabb megközelítést, melyben a szavak mellett más, a spontán beszédre jellemző hangeseeményeket is modelleztünk [11]. Ehhez hasonlóan a diktálási feladat során felmerülő írásjeleket és speciális szimbólumokat is modelleznünk kell, ha hatékonyan szeretnénk őket a felismerőbe integrálni. A problémát érdemes két részre osztani. Egyrészt a kiegészítő funkciót ellátó új szótári elemeket be kell építeni a nyelvi modellbe, másrészt gondoskodni kell az akusztikai szintű modellezésükről is.

4.1 Nyelvi modell felkészítése a diktálási feladatra

A nyelvi modell struktúrájának megváltoztatásához az e-mail felismerő tanítószövegén kell változtatásokat végezni. Elsősorban azt kellett eldönteni, hogy pontosan milyen elemeket is szeretnénk modellezni, és ennek megfelelően kellett átalakítani a tanítókorpusz normalizálását. A kiválasztás során arra törekedtünk, hogy a bevezetett új lexikai elemek segítségével az egyszerűbb elektronikus levelek további kézi kiegészítés nélkül is bevihetőek legyenek. Mint az a **4. táblázatból** is kiolvasható, a legalapvetőbb írásjelek és az „új sor” parancs mellett beépítettünk két emotikont is a nyelvi modellbe, mert úgy ítéltük meg, hogy ezek használata nagyon elterjedt.

4. táblázat: Diktálási szimbólumok a nyelvi modellben.

Felszíni forma	.	!	?	,	\n	:)	:(
Nyelvimodell- szimbólum	<pont>	<fj>	<kj>	<vessző>	<nl>	<mosoly>	<szomorkodás>

Minta a diktáláshoz előkészített tanítószöveg egy sorára:

„a Redmine-on keresetem a VOXerver dokumentációját
<vessző> de végül nem találtam meg <pont> <nl>”

4.2 A diktálási szimbólumok modellezése

4.2.1 Hagyományos megközelítés

A 4. táblázatban bemutatott új szimbólumok akusztikai szintű modellezésére a legelterjedtebben használt megoldás, hogy egy meghatározott hangsorozatra képezzük le őket. A mi rendszerünkben beépített leképezéseket az **5. táblázatban** foglaltuk össze. Nyilvánvaló előnye a megközelítésnek, hogy nagy pontossággal lehet ilyen módon a diktálási szimbólumokat detektálni, amit ki is használ a legtöbb ma forgalomban lévő automatikus diktálórendszer. Nem mehetünk el azonban szó nélkül a hátrányai mellett sem. A diktálás során kényelmetlenséget jelent, hogy minden írásjelet ki kell ejtünk. A felhasználók számára ez egyáltalán nem természetes, hiszen így a rendszer használata gyakorlást igényel, sőt véleményünk szerint egyes felhasználókat pont ez a fajta kényelmetlenség tart távol a diktálórendszerek használatától.

5. táblázat: Diktálási szimbólumok a nyelvi modellben.

Felszíni forma	.	!	?	,	\n	:)	:(
Kiejtett alak	p-o-n-t	f-e-l -k-i-á-l-t-ó -j-e-l	k-é-r-d-ő- j-e-l	v-e-sz-ő	ú-j-s-o-r	m-o-s-o-j	Sz-o-m-o-r -k-o-d-á-s

4.2.2 Prediktív megközelítés

A problémát jobban megvizsgálva észrevehetjük, hogy vannak olyan írásjelek, melyeket önmagában a nyelvi modell képes lehet hatékonyan előre jelezni anélkül, hogy kiejtett alakjukat be kellene diktálni. Ilyen lehet, a „vessző”, mondatzáró „pont” és bizonyos esetekben a „kérdő- és felkiáltójelek”. Az „új sor” parancs és az emotikonok használata sokkal kevésbé szabályokhoz kötött, így ezek detektálása csak a hagyományos módszerrel képzelhető el hatékonyan. Kísérleti rendszerünkben azonban az összes diktálási szimbólumot megkíséreljük a nyelvi modellre támaszkodva detektálni, melynek érdekében akusztikai szinten az összeset semmi vagy szünet (**sp**) modellre képezzük le.

4.3 Kísérleti eredmények

A fejezetben található eredmények a 3.2-es pontban kapott rendszer továbbfejlesztésével jöttek létre.

4.3.1 Hagyományos megközelítés

A hagyományos megközelítés kiértékeléséhez felhasznált tesztfelvételekben az 5. táblázatban bemutatott összes szimbólum bemondásra került kiejtett alakjuknak megfelelő formában. Kísérleteink várakozásainknak megfelelően azt mutatták, hogy ezzel a megközelítéssel a diktálási szimbólumok közel tökéletes pontossággal felismerhető-

ek, miközben a normál szavakra számított hiba sem növekedett meg szignifikáns mértékben. A helyesen felismert szimbólumok aránya (**Corr.** – Correct Rate) átlagosan **93,1%**-os volt.

4.3.2 Prediktív megközelítés

Prediktív megközelítésünk tesztelésének célja elsősorban az volt, hogy kiderítsük, mely diktáláskor fontos lexikai elem felismerését érdemes a nyelvi modellre bízni, és így egyszerűsíteni a diktálást. Tesztelési célokra itt a felvételek egy olyan változatát használtuk, melyben semmiféle diktálási szimbólum nem jelenik meg kiejtett formájában. A kapott eredményeket a **6. táblázat**ban mutatjuk be. A táblázatban csak a „vessző” és „pont” szimbólumok eredményeit tüntettük fel, ugyanis a többi szimbólumra nem kaptunk értékelhető eredményt. A helyesen felismert szimbólumok aránya a „vessző” esetén majdnem 73%-os, azaz a vesszők közel háromnegyedét képes helyesen detektálni a prediktív rendszer. A nem elhanyagolható mértékű beszúrási hiba figyelembevételével is azt mondhatjuk, hogy az automatikus „vessző” detekció beépítése megfontolandó végső rendszerünkbe. A „pont” esetében ugyanez már nem mondható el. Mindössze minden tizedik mondatvégi pontot sikerült helyesen beilleszteni, ami egyelőre nem teszi lehetővé ennek a funkciónak a használatát. Mindezek mellett jó hír, hogy a diktálási szimbólumok beépítése csak minimális hatással volt a többi szó felismerési hibájára. A 3.2-es pontban ismertetett rendszerhez képest mért kevesebb mint 3%-os relatív hibaarány csökkenés elhanyagolhatónak tekinthető.

6. táblázat: A prediktív megoldással kiegészített felismerő kiértékelési eredményei.

	<vessző>		<pont>		WER* [%]
	Corr. [%]	WER [%]	Corr. [%]	WER [%]	
Prediktív megközelítés	72,9	58,9	10,8	92,3	26,7

*A diktálási szimbólumok kivételével az összes szón számolt szóhiba-arány

5 Összefoglalás

Cikkünkben bemutattuk egy olyan, magyar nyelven egyedülálló diktálórendszer fejlesztésének lépéseit, mely akár a felhasználó eszközén futva, egyes írásjelek automatikus elhelyezése mellett képes számok, emotikonok, nagybetűs szavak és rövidítések felismerésére is. Első lépésben a kezdeti rendszerünket ismertettük, melynek hibaarányát a tanítószöveg ékezetesítésével és a modellek személyre szabásával 33%-kal sikerült csökkenteni. Ezután a diktáláshoz szükséges kiegészítő elemek beépítési lehetőségeit vizsgáltuk meg. A legfontosabb írásjelek mellett a soremelés funkciót és a két leggyakrabban használt emotikon felismerését is lehetővé tettük rendszerünkben. Kísérleteink alapján elmondható, hogyha a hagyományos megközelítést követve parancsszavakat rendelünk ezekhez az elemekhez, a detekciójuk minimális felismerési hiba mellett biztosítható. Hátrányként jelentkezik azonban az állandó bemondásukkal járó kényelmetlenség. Ennek kivédése érdekében kísérletet tettünk a diktálást segítő lexikai elemek automatikus észlelésére. Eddigi eredményeink azt mutatják,

hogy a nyelvi modell alapján csak a „vessző” kiváltására kapunk megfelelően pontos előrejelzést, ami érthetővé teszi, miért nem jelentek meg még effajta megoldások az ipari rendszerekben. Véleményünk szerint igény ugyanakkor lenne rá, így ez továbbra is érdekes kutatási terület marad.

További vizsgálataink középpontjában a prediktív írásjel-detekciót helyezzük. Meg kívánjuk vizsgálni, hogy a „vessző” automatikus elhelyezésekor keletkező hiba a gyakorlatban mennyire tolerálható, illetve lehetővé kívánjuk tenni, hogy az automatikus beszúrás mellett normál bemondással is elhelyezhessünk vesszőt. Ezen kívül további kényelmi funkcióként a köszönési és az aláírás formátum személyre szabhatóságát is meg szeretnénk oldani.

Köszönetnyilvánítás

Kutatásunkat a Mindroom (KMOP-1.1.3-08/A-2009-0006), Paelife (AAL-08-1-2011-0001) és a BelAmi (OMFB-00736/2005 BELAMI_H) projektek támogatták.

Hivatkozások

1. Szarvas, M., Furui, S.: Evaluation of the stochastic morphosyntactic language model on a one million word Hungarian task. In: EUROSPEECH2003 (2003) 2297–2300
2. Vicsi, K., Velkei, S., Szaszák, Gy., Borostyán, G., Teleki, C., Tóth, S. L., Gordos, G.: Középszótár, folyamatos beszédfelismerő rendszer fejlesztési tapasztalatai. In: II. Magyar Számítógépes Nyelvészeti Konferencia (2005) 348–359
3. Tarján, B., Mihajlik, P., Balog, A., Fegyó, T.: Evaluation of lexical models for Hungarian Broadcast speech transcription and spoken term detection. In: 2nd International Conference on Cognitive Infocommunications (CogInfoCom) (2011) 1–5
4. Trón, V., Gyepesi, Gy., Halácsy, P., Kornai, A., Németh, L., Varga, D.: Hunmorph: Open Source Word Analysis. In: Proc. of the ACL Workshop on Software (2005) 77–85
5. Chen, S. F., Goodman, J.: An empirical study of smoothing techniques for language modeling. *Computer Speech & Language*, Vol. 13, No. 4 (1999) 359–393
6. Stolcke, A.: SRILM – an extensible language modeling toolkit. In: Proceedings of the International Conference on Spoken Language Processing (2002) 901–904
7. Young, S. J., Evermann, G., Gales, M. J. F., Hain, T., Kershaw, D., Moore, G., Odell, J., Ollason, D., Povey, D., Valtchev, V., Woodland, P. C.: *The {HTK} Book*. Version 3.4. Cambridge, UK: Cambridge University Engineering Department (2006)
8. Mauuary, L.: Blind equalization for robust telephone based speech recognition. In: Proc. of the European Signal Processing Conference (1996) 359–363
9. Mohri, M., Pereira, F., Riley, M.: Weighted finite-state transducers in speech recognition. *Computer Speech and Language*, Vol. 16, No. 1 (2002) 69–88
10. Zainkó, Cs., Csapó, T. G., Németh, G.: Special speech synthesis for social network websites. In: *Lecture Notes in Computer Science* 6231 (2010) 455–463
11. Sárosi, G., Tarján, B., Balog, A., Mozsolics, T., Mihajlik, P., Fegyó, T.: On Modeling Non-word Events in Large Vocabulary Continuous Speech Recognition. In: 3rd International Conference on Cognitive Infocommunications (CogInfoCom) (2012) 649–653