An ASR System for Spontaneous Urdu Speech

Agha Ali Raza	Sarmad Hussain	Huda Sarfraz	Inam Ullah	Zahid Sarfraz
ali.raza[*]	sarmad.hussain[*]	huda.sarfraz[*]	inam.ullah[*]	zahid.sarfraz[*]
NUCES [**]	NUCES	NUCES	NUCES	NUCES

Abstract

One of the major hurdles in the development of an Automatic Spontaneous Speech Recognition System is the unavailability of large amounts of transcribed spontaneous speech data for training the system. On the other hand transcribed read speech data is available comparatively easily. This paper explores the possibilities of training a spontaneous speech recognition system by using a mixture of read and spontaneous speech data. A single speaker, medium vocabulary spontaneous speech recognition system for Urdu has been developed.

1. Introduction

Center for Research in Urdu Language Processing (CRULP; www.crulp.org)[1], in collaboration with Carnegie Mellon University, is working on a project entitled *Telephone-based Speech Interfaces for Access to Information by Non-literate Users*. The goal of this project is to develop speech resources that can be used in the production of dialog systems, enabling users to access online health related information in Pakistan. A speech recognition system for spontaneous Urdu Speech is also developed using these resources.

The HMM based Large Vocabulary Automatic Speech Recognition (LVASR) system for spontaneous Urdu speech is developed using Sphinx 3 [1] trainer and decoder. The training data required for the system is divided into two major categories: a phonetically rich sentence based corpus read out by native speakers of Urdu to provide the continuous speech data, and spontaneous conversational data from recorded interviews of native speakers.

This paper describes the process employed in the training and testing of the speech recognition system. The read and spontaneous speech data are mixed together in various ratios and the system is tested using spontaneous speech data only. The next section briefly reviews similar work done for other languages and the

---

[*] @nu.edu.pk

[**] National University of Computer and Emerging Sciences, Pakistan (www.nu.edu.pk)

phonetic characteristics of Urdu. Section 3 discusses the speech corpus in detail. Sections 4 and 5 describe the processes employed in the development of the phonetic lexicon and in adapting the Urdu ASR system to the Sphinx interface, respectively, along with the tools developed to facilitate these procedures. Section 6 explains the training process and test setup and section 7 analyzes the test results. Finally section 8 presents our conclusions.

2. Background and Literature Review

The task of speech recognition includes the development of speech corpora and phonetic lexicon and training, testing and tweaking of the speech recognition system for the target language.

The process employed in speech corpora development for spontaneous and read speech has already been discussed in detail in [2]. Two essential constraints on a speech corpus are phonetic cover [3] and phonetic balance. Phonetic cover means that the corpus contains all the phones present in the target language and phonetic balance implies that these phones occur in the same relative proportions as in the language itself ([3], [4] and [5]).

The phonetic cover can be phone-based or context-based [6]. Furthermore, the context-based methods can be either diphone [7] or triphone ([8] and [3]) based. However, catering for triphones may not necessarily provide improved accuracy of recognition over diphone based context coverage as shown in [9].

Speech corpora can be developed for different levels of fluency of speech like isolated words (e.g. [10]), continuous speech (e.g. [7], [8] and [11]) and spontaneous speech (e.g. [12], [6] and [3]). Data for the corpus can be gathered using greedy algorithms to maximize the number of sound units in a minimal data set ([7], [13] and [9]) or by developing phonetically balanced sentences from the scratch [5]. The dataset can be made richer by adding transcribed spontaneous speech data [3].

Various techniques are used for improving the performance of speech recognition systems. For spontaneous speech, some of the techniques are (i) to target the frequently mispronounced words and phones and to model them separately ([14], [15]), (ii) to detect

and correctly recognize pauses, word lengthening and filled pauses, in speech [16], (iii) to classify and model the speech disfluencies such as hesitations, repetitions and sentence restarts ([17], [18]), by the use of, e.g., Weighted Finite State Transducers (WFST) [19]. For read speech, like broadcast news, various methods are used to generate sufficient training data including unsupervised approaches like [20] where raw (untranscribed) acoustic data can be transcribed by using an already trained speech recognition system. With languages using Arabic script, e.g. Urdu, Persian, Pashto, Arabic, etc., additional constraint is that the diacritics used to specify vowels are optional and generally now written in text. One solution to this problem is to train the initial models using fully manually diacritized transcribed speech [21]. Then using these initial training models unsupervised learning of the non transcribed data is performed.

In terms of Word Error Rate (WER) the systems range from accurate systems like [21] used for the transcription of Arabic broadcast news with a WER of 14.9% to spontaneous microphone based systems like [19] for paraphrasing spontaneous Japanese speech into written style sentences with a WER of 24% and telephone based spontaneous speech recognition systems with WERs around 29% [17]. These systems are for multiple speakers.

3. Speech Corpora

As discussed, our aim has been to develop LVASR for Urdu. Speech corpora for Urdu is not available and thus had to be collected. Complete details of the text corpus design and spontaneous speech data acquisition process has been reported earlier in [2]. The characteristics of speech corpora that was used for training and testing this system, is summarized below.

3.1. Read Speech Corpus

This corpus consists of 70 minutes of transcribed read speech consisting of 708 greedily made sentences representing all the phones and triphone combinations[2] in Urdu. The data consists of 10101 tokens with 5656 unique words. It contains 60 unique phones and 42289 phone occurrences. It must be noted that the sentences contained in this corpus are all hand made by trained linguists following the greedy approach to accommodate all the words (which were selected by a set cover algorithm) and to prevent additional words as much as possible [2]. Therefore, while correct grammatically, there are some instances where these sentences are not semantically normal.

---

[2] Some contexts were collapsed due to similarity, to control the number of combinations. See [2] for further details.

Figure 1 shows the comparison of the frequencies of occurrence of the Urdu phones in the corpus and the frequencies of occurrence of the same phones in the sentence list. Figure 2 shows the logarithmic plot of frequency of occurrence of each tri-phone in the corpus (the curve above) and its frequency of occurrence in the sentences (shown below). Therefore, the data is phonetically balanced and also provides complete tri-phonemic cover as shown in Figure 1 and Figure 2.
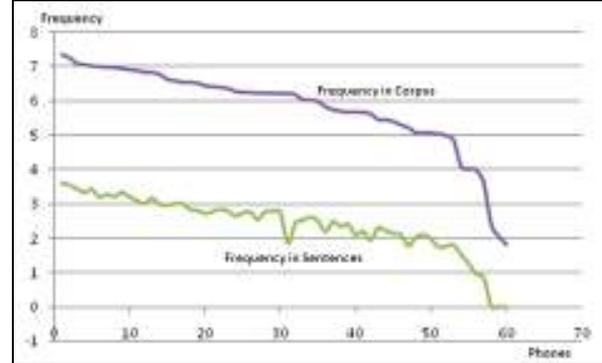

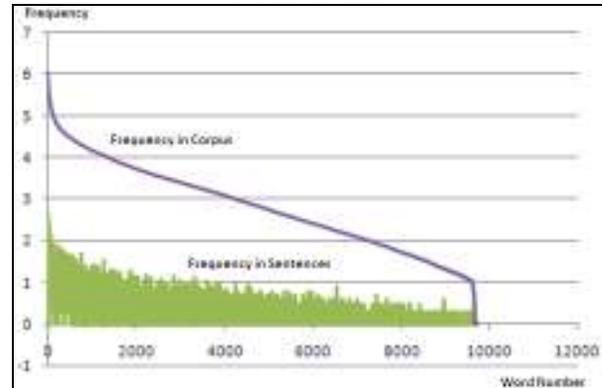**Figure 1 Phone frequencies ($\log_{10}$) in the corpus vs. the sentences**

Figure 2 Triphoneme frequencies (log10) in the corpus vs. the sentences [2]

3.2. Spontaneous Speech Corpus

The spontaneous corpus consists of 109 minutes of transcribed spontaneous speech consisting of 3266 utterances recorded in the form of interviews [2]. The data consists of 21034 tokens with 2032 unique words. It contains 60 different phones with 72700 phone occurrences. This data represents the natural and spontaneous speech patterns of a native speaker of Urdu.

The combined data from the spontaneous and read (excluding 22 minutes of spontaneous speech data, which is used only for testing purposes) contains 3174 utterances spanning over 157 minutes of speech. It contains 31135 word tokens with 6693 unique words and 114990 phone tokens (62 unique phones including the rare /l^h/).

3.3. Recording Environment

The recordings are carried out in multiple sessions (of 15-30 minutes duration), in normal home and office environment with ambient background noise (e.g. from fans, air conditioners etc. if these continue during the entire length of one session). Intermittent noise of any kind (e.g. coughing, opening and closing of doors) was not allowed and any overlapping phrases were rerecorded.

4. Lexicon Development

One of the major questions was regarding phonetic vs. phonemic transcription for the lexicon. The later is mostly rule-based and can follow from the Urdu script without even hearing the actual speech (e.g. see [22]). On the other hand, the phonetic (or narrow) transcription requires inspection of the actual speech files. This is time consuming and poses additional challenges as sometimes the uttered phones may actually lie on the boundary between two phones and an objective decision is not simple. The process may require the study of spectrograms to reveal the actual phone designation in addition to perceptual response. However, this approach establishes a more accurate mapping between phones and acoustic waveforms.

It was decided that the initial training of the speech recognition system would be done on the basis of phonemically transcribed corpora. This will rapidly generate the test results and then on the basis of the error analysis the corpus can be phonetically transcribed in part or as a whole at a later stage. However, diacritics based disambiguation was done for confusable words of Urdu to facilitate correct lookup of the phonemic lexicon.

4.1 Tools Developed – The Urdu Auto completer and Lexicon Development Utility

In order to facilitate the task of transcription of the interviews and building of the lexicon an Urdu auto completion and phonetic transcription utility was developed. The main features of the utility are to give word auto complete suggestions and to provide letter to sound based phonemic transcription [22] suggestions for new words. The main objectives of the utility were as follows:

- To facilitate the task of Urdu transcription by providing auto complete options from the lexicon
- To prevent spelling errors
- To allow the typist to write the words in exactly the same way as previously available in the lexicon. This prevents errors at a later stage when these transcriptions are compiled for use with Sphinx. This is necessary as many words of Urdu can be correctly written using more than a single

way (e.g. with or without diacritics, or even with partial diacritics)
- In addition, this will prevent or at least reduce Unicode normalization errors
- To indicate that a typed word does not exist in the lexicon and thus, has to be added also allowing smooth addition of new entries to the lexicon
- To allow phonetic transcription of words in CISAMPA format (Section 5.1)
- To give the facility of rule based letter to sound conversion of Urdu words

5. Adaptation of Urdu corpora to ASCII based Sphinx ASR interface

The Sphinx speech recognition system requires many different files in specific formats to be able to perform the training and decoding tasks. Manually generating these files is a lengthy job which is also more susceptible to errors, which may not be easy to detect at a later stage. Therefore a compilation utility was developed which generates all the files required by Sphinx for training and testing the ASR system using Unicode based Urdu files as input.

5.1 Phonetic transcription using CISAMPA

The phonemic transcription needs to be done using some notation. IPA uses the Unicode character set and is hence not usable as Sphinx-3 does not support Unicode. The SAMPA character set had to be abandoned as Sphinx-3 is not case sensitive; whereas SAMPA distinguishes between many characters on the basis of case like n (for [n]) vs. N (for [ŋ]). X-SAMPA [23] could not be used as it largely relies upon special characters in its character set e.g. \, < etc. These characters cannot be used as files names (as was required in our work) and moreover certain software systems treat these special characters as control characters or position markers.

ARPABET [24] could have provided the solution but the ARPABET notation is too specifically designed for American English pronunciation and is difficult to read for Urdu sounds. For example, Urdu word بجلی ([b ɪ ʤ l i]) is more readable as B I D_ZZ L II (in CISAMPA) than B IH JH L IY (in ARPABET) or بَڑا ([b ∂ ɽ a]) can be represented as B A RR AA (in CISAMPA) but we were unable to find any character for the retroflex [ɽ] in ARPABET; same is the case for many other Urdu specific sounds, like nasal vowels. In short, ARPABET is English specific and not suitable for Urdu.

Therefore, a case insensitive notation free from special characters was required. Therefore, using SAMPA character set as a starting point a phonetic character set was developed which was named

CISAMPA (Case Insensitive SAMPA) (Appendix-A). The basic rules of conversion from SAMPA to CISAMPA are as follows:

- The complete character set is written in capital case (but is case insensitive)
- The character set does not include any punctuation mark or special character like @ or / or ? etc.
- Most of the consonants have been converted simply by converting them into capital form
- Dentals are indicated by an _D, as [t̪] is represented as T_D
- Aspiration is indicated by _H, as [t̪ʰ] is represented by T_D_H
- Retroflex is indicated by double characters e.g. [ʈ], [ɖ] and [ɽ] are represented as TT, DD and RR (this remains true for the alveolar versions of the former two as well i.e [t] and [d])
- Short vowels are indicated by single capital character while long ones by double capital characters (A for [ə] and AA for [ɑ])
- Some vowels are represented by ARPABET like notations like [e] is represented as AE
- Nasals are represented by appending an N e.g. [ẽ] is represented as AEN

5.2 Unicode text format

As mentioned earlier, Sphinx-3 does not support Unicode text format, while Urdu script uses Unicode characters. Therefore, a Unicode to ASCII mapping mechanism was developed. As the phonemic transcription in CISAMPA is done using the lexical lookup, and the CISAMPA notation is completely ASCII based therefore the Romanization is simply done by removing the spaces from the CISAMPA transcription. This produces a one-to-many mapping between Urdu and CISAMPA but a one-to-one mapping between Romanization and CISAMPA.

5.3 Tools developed – the Sphinx Compiler

If all the files required by Sphinx are generated manually, it will be a very time consuming task and will result in a lot of errors. Therefore, it was required to automate this process as much as possible. An application was developed for the generation of these files and for the analysis of training and test data as well.

6. Test Setup

Experiments were devised with the goal of finding the optimal spontaneous-to-read data ratio that would give best recognition results on spontaneous speech. The experiments involve 87 minutes of spontaneous speech training data and 70 minutes of read speech training data. The system is tested with 800 utterances

(22 minutes) of spontaneous speech (non-overlapping with the training data). The system is then progressively trained with 100% of spontaneous speech + x % of read speech (where x increases in steps of 25 % from 0 to 100% of available read speech). Next the system is trained with a mixture of 100% read speech + x% of spontaneous speech (where x increases in steps of 25% from 0 to 100% of available spontaneous speech). All other parameters are kept constant to observe the required trend only. The statistics of the two types of training data and the test data are mentioned in Table 1.

	Spontaneous Training Data	Read Training Data	Spontaneous Test Data
No. of utterances	2466	708	800
Duration (minutes)	87	70	22
No. of words	21034	10101	4623
No. of uniq. words	2032	5656	750
No. of Phones	72700	42289	16442
No. of uniq. Phones	60	60	55

**Table 1 - Training and Test Data for Sp:Re Ratio Experiments**

The experiments were performed using language models derived from the actual training data. Therefore, the LM varies from test to test as the ratio between the spontaneous and read speech varies in the training data. The LM in all cases is a trigram language model with Witten-Bell discounting generated using the SLM Toolkit.

7. Test Results

The recognition results are shown in Table 2 along with the details regarding number of unique out of vocabulary words (OOVs), the number of instances of OOVs in the test data.

Training Data (Spontaneous:Read)	WER Training LM	OOVs	OOV Instances	LM OOVs
100:0	22.9	212	471	212
100:25	21.5	182	410	182
100:50	21.0	168	347	168
100:75	20.3	154	324	154
100:100	18.8	136	279	136
75:100	22.1	151	329	151
50:100	23.7	174	384	174
25:100	29.1	209	445	209
0:100	58.4	297	826	297

**Table 2 - Results with Training Data based Language Model**
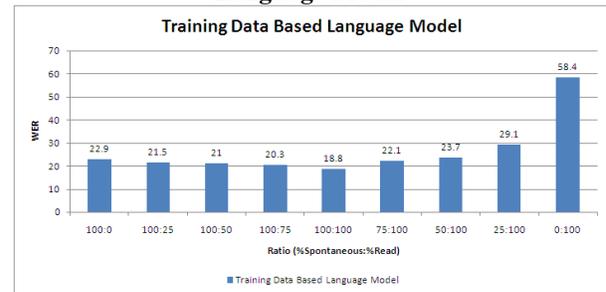
Figure 3 - Results with Training Data based Language Model
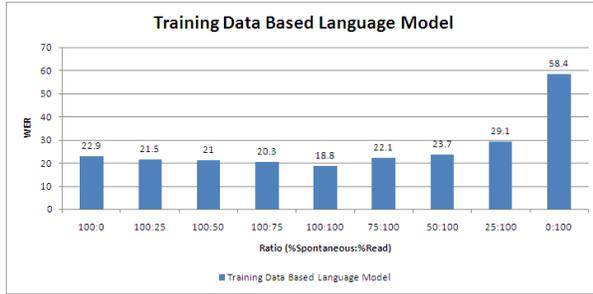
Figure 3 - Results with Training Data based Language Model

These results are graphically summarized in Figure 3. Figure 4 depicts the relationship between WER and OOVs for different training ratios. In all tests the beam width used is 1e-700 and language weight is 8. Hence, all other factors except the spontaneous to read ratio are maintained constant.

The results clearly depict the effects of the ratio on recognition results. It can be seen that the WER starts decreasing as the read speech is introduced into the mixture of training data hence increasing the overall amount of data as well. The WER reaches a minimum of 18.8% for the 1:1 ratio between spontaneous and read speech and then begins to climb rapidly as the spontaneous data becomes limited in the mixture. Finally reaching a high WER of 58.4% for read speech based training data. The results also indicate that the system is still in need of more training data in term of duration and amount as we can see that the least WER is obtained for the maximum amount of training data.
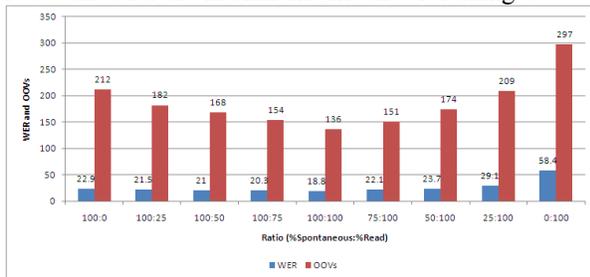
Figure 4 - Comparison of WER with OOVs in the Test Data

Figure 4 shows the relationship between OOVs and WER. The correlation between WERs and OOVs gives a very high 0.92 and that between WERs and OOV instances gives 0.96 (Table 2). It is also notable that OOVs also mean that the language model does not contain those words either. The sharp peak of WER 58.4% corresponds with the highest OOV value. The high correlation between word error rates and out of vocabulary words indicates that a further decrease in WER may be achievable by decreasing the OOVs, which can be accomplished by expanding the corpus for training data and language model.

The most interesting result of these tests is perhaps the WER of 29.1% for the spontaneous to read ratio of 25:100. This indicates that even a small addition of spontaneous speech training data in a phonetically balanced read speech training data can produce a satisfactory outcome (compared to the 22.9% WER with spontaneous training data only). However, testing a read speech trained system with spontaneous data (0:100) gives poor results as expected.

8. Conclusion

The work presented in this paper is an attempt to alleviate the issue of scarcity and unavailability of large amounts of transcribed spontaneous speech data for training spontaneous speech ASR systems. This is especially true for languages in which speech resources are not yet available to a large extent. The solution is to train the ASR systems with phonetically balanced and covering read speech data (which is available comparatively easily) and only add a small percentage of spontaneous training data to the mixture to achieve satisfactory results. Furthermore, the techniques developed for this work can benefit other Unicode based languages which can now use ASCII based ASR systems. Finally, the CISAMPA can be a useful phonetic transcription notation for applications where case-sensitivity and use of special characters in the notation can cause problems.

More work is in progress to convert the basic single speaker ASR into a multi-speaker system, to increase the amount of training data and to develop a bigger corpus for generating representative language models.

Acknowledgements

References

[1] "Cmusphinx: The carnegie mellon sphinx project." http://cmusphinx.sourceforge.net/html/cmusphinx.php.

[2] A. Raza, S. Hussain, H. Sarfraz, I. Ullah, and Z. Sarfraz, "Design and development of phonetically rich Urdu Speech Corpus," in *Proceedings of O-COCOSDA'09 and IEEE Xplore*, 2009.

[3] A. L. Ronzhin, R. M. Yusupov, I. V. Li, and A. B. Leontieva, "Survey of russian speech recognition systems,"

[4] S. T. Abate, W. Menzel, and B. Tafila, "An amharic speech corpus for large vocabulary continuous speech recognition," ISCA, 2005. Ninth European Conference on Speech Communication and Technology.

[5] L. Villaseñor-Pineda, M. Montes-y Gomez, D. Vaufreydaz, and J. F. Serignat, "Experiments on the construction of a phonetically balanced corpus from the web," *Lecture notes in computer science*, pp. 416–419, 2004.

[6] A. Li, F. Zheng, W. Byrne, P. Fung, T. Kamm, Y. Liu, Z. Song, U. Ruhi, V. Venkataramani, and X. X. Chen, "Cass: A phonetically transcribed corpus of mandarin spontaneous speech," ISCA, 2000. Sixth International Conference on Spoken Language Processing.

[7] G. Anumanchipalli, R. Chitturi, S. Joshi, R. Kumar, S. P. Singh, R. N. V. Sitaram, and S. P. Kishore, "Development of indian language speech databases for large vocabulary speech recognition systems,"

[8] V. Chourasia, K. Samudravijaya, and M. Chandwani, "Phonetically rich hindi sentence corpus for creation of speech database," *Proc. O-COCOSDA*, p. 132–137, 2005.

[9] Y. C. Yio, M. S. Liang, Y. C. Chiang, and R. Y. Lyu, "Biphone-rich versus triphone-rich: a comparison of speech corpora in automatic speech recognition," pp. 194–197, 2005. Cellular Neural Networks and Their Applications, 2005 9th International Workshop on.

[10] G. Raškinis, "Building medium-vocabulary isolated-word lithuanian hmm speech recognition system," *Informatica*, vol. 14, no. 1, pp. 75–84, 2003.

[11] V. Digalakis, D. Oikonomidis, D. Pratsolis, N. Tsourakis, C. Vosnidis, N. Chatzichrisafis, and V. Diakoloukas, "Large vocabulary continuous speech recognition in greek: Corpus and an automatic dictation system," ISCA, 2003. Eighth European Conference on Speech Communication and Technology.

[12] D. Binnenpoorte, C. Cucchiarini, H. Strik, and L. Boves, "Improving automatic phonetic transcription of spontaneous speech through variant-based pronunciation variation modelling," p. 681–684, 2004. Proceedings of the International Conference on Language Resources and Evaluation (LREC).

[13] B. Bozkurt, O. Ozturk, and T. Dutoit, "Text design for tts speech corpus building using a modified greedy selection," ISCA, 2003. Eighth European Conference on Speech Communication and Technology.

[14] T. Slobada and A. Waibel, "Dictionary learning for spontaneous speech recognition," in *Spoken Language, 1996.*

[15] J. Nedel, R. Singh, and R. Stern, "Automatic Subword Unit Refinement for Spontaneous Speech Recognition Via Phone Splitting," in *Sixth International Conference on Spoken Language Processing*, ISCA, 2000.

[16] M. Goto, K. Itou, and S. Hayamizu, "A real-time filled pause detection system for spontaneous speech recognition," in *Sixth European Conference on Speech Communication and Technology*, ISCA, 1999.

[17] J. Duchateau, T. Laureys, and P. Wambacq, "Adding robustness to language models for spontaneous speech recognition," in *COST278 and ISCA Tutorial and Research Workshop (ITRW) on Robustness Issues in Conversational Interaction*, ISCA, 2004.

[18] V. Rangarajan and S. Narayanan, "Analysis of disfluent repetitions in spontaneous speech recognition," *Proc. EUSIPCO 2006*.

[19] T. Hori, D. Willett, and Y. Minami, "Paraphrasing spontaneous speech using weighted finite-state transducers," in *ISCA & IEEE Workshop on Spontaneous Speech Processing and Recognition*, ISCA, 2003.

[20] F. Wessel and H. Ney, "Unsupervised training of acoustic models for large vocabulary continuous speech recognition," *IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 1, pp. 23–31, 2005.

[21] H. Soltau, G. Saon, B. Kingsbury, J. Kuo, L. Mangu, D. Povey, and G. Zweig, "The IBM 2006 Gale Arabic ASR System,"

[22] S. Hussain, "Letter to Sound Rules for Urdu Text to Speech System," 2004. Proceedings of Workshop on "Computational Approaches to Arabic Script-based Languages", COLING 2004, Geneva, Switzerland.

[23] "X-sampa." http://coral.lili.uni-bielefeld.de/langdoc/-EGA/Formats/Sampa/sampa.html.

[24] "Arpabet and the timit alphabet." http://-www.laps.ufpa.br/aldebaro/papers/ak_arpabet01.pdf.

Appendix A

#	IPA	SAMPA	CISAMPA	#	IPA	SAMPA	CISAMPA	#	IPA	SAMPA	CISAMPA	#	IPA	SAMPA	CISAMPA
1	p	P	P	17	k	K	K	33	h	H	H	49	ð	o~	OON
2	pʰ	p_h	P_H	18	kʰ	k_h	K_H	34	l	L	L	50	ɔ	O	O
3	b	B	B	19	g	G	G	35	lʰ	l_h	L_H	51	ɔ̃	O~	ON
4	bʰ	b_h	B_H	20	gʰ	g_h	G_H	36	r	R	R	52	ɑ	A	AA
5	m	M	M	21	ŋ	N	NG	37	rʰ	r_h	R_H	53	ɑ̃	A~	AAN
6	mʰ	m_h	M_H	22	ŋʰ	N_h	NG_H	38	ɽ	r'	RR	54	i	I	II
7	t̪	t_d	T_D	23	q	Q	Q	39	ɽʰ	r'_h	RR_H	55	ĩ	i~	IIN
8	t̪ʰ	t_d_h	T_D_H	24	ʔ	?	Y	40	j	J	J	56	e	e	AE
9	d̪	d_d	D_D	25	f	F	F	41	jʰ	j_h	J_H	57	ẽ	e~	AEN
10	d̪ʰ	d_d_h	D_D_H	26	v	V	V	42	ʧ	t_S	T_SH	58	ɛ	E	E
11	t	t'	TT	27	s	S	S	43	ʧʰ	t_S_h	T_SH_h	59	æ	{	AY
12	tʰ	t'_h	TT_H	28	z	Z	Z	44	ʤ	d_Z	D_ZZ	60	æ̃	{~	AYN
13	d	d'	DD	29	ʃ	S	SH	45	ʤʰ	d_Z_h	D_ZZ_h	61	ɪ	I	I
14	dʰ	d'_h	DD_H	30	ʒ	Z	ZZ	46	u	u	UU	62	ʊ	U	U
15	n	N	N	31	χ	X	X	47	ũ	u~	UUN	63	ə	@	A
16	nʰ	n_h	N_H	32	ɣ	7	7	48	o	o	OO				