Continuous Speech Recognition Consortium — an Open Repository for CSR Tools and Models —

Akinobu Lee[*], Tatsuya Kawahara[**], Kazuya Takeda[***], Masato Mimura[†], Atsushi Yamada[†], Akinori Ito[‡], Katsunobu Itou[⋆], Kiyohiro Shikano[*]

[*]Nara Institute of Science and Technology
Takayama-cho, Ikoma 630-0101, Japan
{ri,shikano}@is.aist-nara.ac.jp

[**]Kyoto University
Sakyo-ku, Kyoto 606-8501, Japan
kawahara@kuis.kyoto-u.ac.jp

[***]Nagoya University
Chikusa-ku, Nagoya 464-8603, Japan
takeda@nuee.nagoya-u.ac.jp

[†]ASTEM
Simogyo-ku, Kyoto 600-8813, Japan
{mimura, yamada}@astem.or.jp

[‡]Tohoku University
Aoba-ku, Sendai 980-8577, Japan
aito@fw.ipsj.or.jp

[⋆]Agency of Industrial Science and Technology
Umezono, Tsukuba 305-8568, Japan
itou@ni.aist.go.jp

Abstract

Continuous Speech Recognition Consortium (CSRC) was founded on 2000 to promote sharable high-quality platform for research and development of speech recognition. It is a continued work of the former Japanese Dictation Toolkit project from 1997 to 2000. An overview of the software developed in the first year (Oct. 2000 - Sep. 2001) is given in this paper. We have revised the LVCSR (large vocabulary continuous speech recognition) engine Julius, and constructed new acoustic models using very large speech corpora. Moreover, a variety of acoustic and language models as well as tools are being set up. Currently over 50 companies and academic institutes are joining. The software is available by contacting the address `csrc@astem.or.jp`.

1. Introduction

In recent days, the speech recognition technology has attracted much public interest, and various studies for each component (acoustic models, language models and a decoder) has been investigated. However, the current recognition system are not sufficient in accuracy, robustness and usability to realize a truly practical speech interface in real-world environment. To promote further improvement and assessment of each component as well as development of various speech recognition system, a sharable platform that provides an common integration framework of the components is essential.

This background has motivated us to develop a free and open sharable platform that can be used as a baseline and reference system. It is rather easy to have agreement of a common interface and format in the recognition system. In the previous work, a common LVCSR (Large Vocabulary Continuous Speech Recognition) platform, "Japanese Dictation Toolkit", has been developed from 1997 to 2000 by collaborative works of researchers in different academic institutes under the government support (Itou et al., 2000; Kawahara et al., 2000). The toolkit has been distributed to about 250 institutes, and successfully worked as a reference and a baseline system. In the rest of this paper, we refer to the former project as "IPA".

In order to enhance the continued work and make the platform further open to public, an association named "Continuous Speech Recognition Consortium (CSRC)" was founded on Oct. 2000. The aim is to maintain a high-quality sharable software repository and promote further studies and developments of speech recognition systems.

This paper introduces our software and models in the first year (from Oct. 2000 to Sep. 2001). An overview of each component and improvements are shown.

2. Recognition engine Julius

We have revised our open-source LVCSR engine Julius (Lee et al., 2001b). Julius is a tree-trellis based 2-pass speech recognition software capable of real-time, on-the-

fly decoding on most current PCs in 20k-word dictation task. The most remarkable feature is that it is designed to be open. As the search algorithm is independent of word or phone unit, word 3-gram and dictionary of any word or phone units can be used as a language model. For acoustic model, parameter tying in state level, mixture level, codebook level and Gaussian density level are supported. Thus various HMM types can be used including shared-state triphones and tied-mixture models. Any number of mixtures, states, and phone units are allowed. Standard formats are adopted for model interface: ARPA standard format for language model, and HTK format for HMM definition and dictionary. Thus it can use models built by other free software such as CMU-Cambridge SLM toolkit (Clarkson and Rosenfeld, 1997) and HTK (Young et al., 1995). Although Julius has been developed under Japanese dictation task, it is applicable to any other language since the algorithm is independent of word or phone unit. The main development platform is Linux, and the source code is also open.

The recent improvements of Julius focus on faster decoding, extension for spontaneous speech, integration of grammar-based recognition and support for Microsoft Windows. The details are described below.

2.1. Faster computation

To speed up acoustic likelihood computation, a decoding technique called Gaussian Mixture Selection is implemented (Lee et al., 2001a). While recognition, all base phones are first evaluated using a simple monophone HMM, and then only the k-best states are re-computed according to the corresponding triphone HMM. This method can reduce acoustic computational cost farther by about 30%, and makes Julius to achieve over 92% word accuracy in real-time decoding on PCs for 20k dictation task.

2.2. Enhancement for spontaneous speech

Several decoding enhancement are implemented for the purpose of realizing a high accurate recognition of spontaneous speech.

As a break of utterance can not be explicitly determined in spontaneous speech, segmentation of input speech is not easy. Thus, a successive decoding is implemented that performs recognition and segmentation simultaneously. In the first recognition pass of frame-synchronous beam search, the first pass terminates if pause model gets maximum likelihood for a certain period. The second pass is executed for the last segment, and the resulting word hypothesis is kept to the next segment as word context. This technique contributes to an improvement of recognition accuracy in long and randomly paused utterance such as lecture speech.

A transparent word is introduced to handle fillers and pause insertion effectively. When computing N-gram probability, transparent words are skipped as a word context to avoid affecting nearby words. This method is effective if a training text contains no filler words and probabilities of fillers are not well estimated.

2.3. Grammar-based engine Julian

Although statistical language models such as word N-gram are commonly used in current recognition systems, written grammar based recognition is still of great use in small tasks and sometimes easier to handle than corpus-based statistical models. Thus, a grammar-based recognition parser Julian, formerly developed in Kyoto University, is integrated to Julius.

The format of grammar is an original one. It has a concept of word category and should be given in separate files. Rewrite rules should be written in BNF using word category names as terminal symbols, and actual recognition words per category are defined in dictionary file with their pronunciations. As Julian treats the grammatical constraint in definite finite state automaton internally for efficiency, BNF grammars should be compiled to automaton network before recognition. The class of grammar is checked on the compilation time if it does not exceed regular expression class. The grammar compiler, several tools to develop a grammar and sample grammars are included in the distribution.

The decoding algorithm of Julian is a conventional two-pass A* search based on finite automaton network with word-pair constraints as heuristics. Most of model interfaces and decoding parameters are the same in Julius and Julian. Actually, both engine shares most of the source codes. They are distributed in a single source tree and can be switched at compilation time.

2.4. Windows and SAPI support

The target users of our former project was researcher and developers engaging in speech recognition, and Unix OS was the main environment. To promote our work further and encourage development of speech application such as multi-modal interface or multimedia processing, an application programming interface should be provided.

Microsoft Windows version of Julius / Julian is implemented with Microsoft Speech API 5.1 support. Now applications can control Julius via SAPI on Windows. However, current implementation is not fully compliant to SAPI requirements. SAPI-style grammar format (XML), multi-context and multi-instance grammar are not supported yet. The recognition performance of Windows version is as same as Unix version.

An another Windows version using DLL interface is also developed by a consortium member. It is also available for free, and will be included in the next 2001 version of the toolkit.

3. Statistical language model

In the former IPA project, we offered a generic word 3-gram language model trained by Mainichi newspaper articles of 75 months. The refined version of the newspaper model is offered. A morphological analyzer ChaSen is used for all models to split undelimited Japanese training text into morphemes. The format is either in binary format of CMU-Cambridge SLM toolkit or ARPA standard format.

3.1. Japanese newspaper model

Japanese 20k and 60k word 3-gram language model trained from newspaper article corpus is refined. The training database is extended to 111 months of Mainichi news-

paper articles (from Jan. 1991 to Jun. 2000, excluding 3 months in 1994 for test set). The cut off parameter was 1 for 2-gram and 3-gram, and Witten Bell discounting is used for back-off smoothing.

3.2. Tools for N-gram language modeling

Several tools are provided to help development of word N-gram language model. All tools are open source softwares donated by the members.

palmkit[1]: an N-gram training tool fully compatible with the CMU-Cambridge SLM Toolkit in command level. It supports class N-gram and N-gram count mixture as well as combined language model using linear interpolation. As the language model combination is supported within API level, the SLM library in this toolkit enables any tool to exploit the LM combination.

mergelm: this tool merges two different N-gram model into one. Unlike other merging tools, it performs a "complementary back-off" that properly estimates back-off probabilities of unseen N-gram entries from another model. It does not need extra information about source training corpus, and capable of merging ARPA format N-gram models directly.

webcollect: an automatic text collector using Web search engine. Given a keyword, it searches for Web sites using search engines and collect the relevant Web pages. Statistical garbage filtering based on character perplexity is incorporated to distinguish non-text part from the HTML files (currently for Japanese only).

LMCompress: 3-gram compressor that reduces 3-gram entries according to their entropy. Experiments showed that 3-gram entries can be reduced to only 10% without loss of recognition accuracy.

4. Acoustic model

In the IPA toolkit, Japanese acoustic models trained with ASJ speech databases of phonetically balanced sentences (ASJ-PB) and newspaper article texts (ASJ-JNAS) (Itou et al., 1998) were provided. In the consortium, the training data is extended to cover various speech targets.

Larger-scale Japanese acoustic model (CSRC model) is built to make a high-standard generic acoustic model. The phone balanced set from ATR spontaneous speech database (Matsui et al., 1999) (ATR/BLA) is added to the training corpora. The outline of the training data is shown in Table 1. Training set is now extended to have 4,130 speakers, 169,348 sentence, total 260 hours. It is 11 times as much in number of speakers, and 2.6 times as much in quantity.

The trained models are listed in Table 2. They are all gender-independent models. The largest model has as much as three hundred thousand Gaussian densities, larger by six times as large as the IPA model. In addition to a context-independent monophone model and shared-state triphone model, PTM (phonetic tied-mixture) triphone model (Lee et al., 2000) are also trained. It is a middle-class model between monophone and shared-state triphone. 129

[1] `http://palmkit.sourceforge.net/`

Table 1: Outline of training data for CSRC model

		# speaker	# sentence	hour
ATR/BLA	male	1379	42057	59h
	female	2390	70483	103h
	total	3769	112540	162h
ASJ/JNAS + ASJ/PB	male	179	28127	47h
	female	182	28681	51h
	total	361	56808	98h
total	male	1558	70184	106h
	female	2572	99104	154h
	total	4130	169348	260h

Table 2: CSRC models

	# state	# mixture
monophone	129	8, 16, 32, 64, 128
triphone 2000	2000	6, 16, 32
triphone 5000	5000	8, 16, 32, 64
PTM triphone	3000/129	64, 128

Gaussian codebooks are prepared for each state of monophone, and shared among 3000 triphone states with different mixture weights.

Several target-dependent models are also made to cover various speech. An experimental telephone-band model is provided. This model is trained by the ASJ-JNAS corpus, with limiting the band width of training data from 300Hz to 3400Hz. Gender-independent shared-state triphone that has 2,000 states and 16 or 32 mixtures are provided.

An acoustic model for elderly person is also donated. The model is trained by elderly speech database that consists of 301 speakers from age of sixty to ninety, and each person read 100 phone-balanced sentences and 100 newspaper articles (Baba et al., 2001). Monophone and PTM triphone model are included.

All models are three-state continuous density HMM with diagonal covariance Gaussian mixtures in HTK format. The phone set and speech analysis parameters are the same as the IPA toolkit.

5. Evaluation

Evaluation of the CSRC acoustic models are carried out. To compare the performance with the former IPA models, the same test set of newspaper reading task is taken. It consists of 200 sentences by 26 speakers, equal number of male and female. Another test set on travel arrangement task is also prepared to evaluate their performance on spoken dialogue. It contains 212 sentences from ATR speech database (ATR/SDB) (Nakamura et al., 1996) and 108 sentences from ATR speech and language database (ATR/SLDB) (Morimoto et al., 1994). The latter set has of only male speakers.

Recognition accuracy using IPA model and new CSRC model for each test set is shown in Table 3 and Table 4. On 20k-word read task (IPA-98-testset), the largest model of 5000x64 Gaussians showed the best accuracy of 95.37%,

Table 3: Word accuracy of IPA model trained by ASJ (%)

		monophone	PTM	triphone
	# state		3000	2000
	# mixture	16	64	16
IPA	male	82.07	90.55	92.45
	female	84.03	94.79	94.79
ATR/SDB	male	63.82	75.25	72.50
ATR/SLDB	male	78.70	88.94	88.54

Table 4: Word accuracy of CSRC model trained by ASJ+ATR/BLA (%)

		monophone				PTM	triphone				
	# state					3000	2000		5000		
	# mixture	16	32	64	128	64	16	32	16	32	64
IPA	male	78.58	78.58	80.77	85.16	90.17	92.46	91.95	91.63	93.60	94.61
	female	86.25	88.76	88.76	91.43	93.97	94.22	94.36	94.80	94.54	96.13
ATR/SDB	male	62.80	65.04	68.52	68.65	75.69	74.24	75.24	75.54	76.00	76.19
ATR/SLDB	male	82.82	83.35	85.63	85.91	90.42	90.52	91.61	89.55	90.50	91.14

that is the best figure ever reported in the test set. As the IPA model was saturated at 2000x16 Gaussians, the extended training corpus enables more large-scale model to be trained fairly. Accuracy of male speakers were relatively inferior to female speakers. This may be caused by the imbalanced number of male and female speakers in ATR/BLA.

On dialogue task, CSRC model showed better accuracy than IPA model. Thus the effect of using large-scale training corpus and dialogue corpus is practically confirmed.

6. Conclusion

The CSRC (Continuous Speech Recognition Consortium) software and models in the first year are introduced. This toolkit has open interface between modules and models, and all softwares are open with source codes to keep generality and scalability for researchers and developers. This platform has a capability to work as a reference or a baseline system.

All the products and tools are available to the CSRC members. Currently over 50 companies and academic institutes join in this project. This software is available by contacting the address `csrc@astem.or.jp`.

We plan to release our next distribution of 2001 version on Oct. 2002. Currently, the following issues are planned to be included in the next distribution.

- English distribution of Julius/Julian with English documentation and sample models.
- Further extension of Julius/Julian: dynamic grammar switching, support for XML grammar, more integrated and refined API, etc.
- Extensive Japanese language model and acoustic model trained by numerous corpora.
- Improved language models for spontaneous speech.
- Acoustic models for various environment including car environment model and child voice model.

7. References

A. Baba, S. Yoshizawa, M. Yamada, A. Lee, and K. Shikano. 2001. Elderly acoustic model for large vocabulary continuous speech recognition. In *Proc. EUROSPEEECH*, pages 1657–1660.

P. R. Clarkson and R. Rosenfeld. 1997. Statistical language modeling using the cmu-cambridge toolkit. In *Proc. EUROSPEECH*, volume 5, pages 2707–2710.

K. Itou, M. Yamamoto, K. Takeda, T. Takezawa, T. Matsuoka, T. Kobayashi, K. Shikano, and S. Itahashi. 1998. The Design of the Newspaper-Based Japanese Large Vocabulary Continuous Speech Recognition Corpus. In *Proc. ICSLP*, volume 7, pages 722–725.

K. Itou, K. Shikano, T. Kawahara, K. Takeda, A. Yamada, A. Itou, T. Utsuro, T. Kobayashi, N. Minematsu, M. Yamamoto, S. Sagayama, and A. Lee. 2000. IPA Japanese dictation free software project. In *Proc. LREC*, pages 1343–1349.

T. Kawahara, A. Lee, T. Kobayashi, K. Takeda, N. Minematsu, S. Sagayama, K. Itou, A. Ito, M. Yamamoto, A. Yamada, T. Utsuro, and K. Shikano. 2000. Free software toolkit for Japanese large vocabulary continuous speech recognition. In *Proc. ICSLP*, volume 4, pages 476–479.

A. Lee, T. Kawahara, K. Takeda, and K. Shikano. 2000. A new phonetic tied-mixture model for efficient decoding. In *Proc. IEEE ICASSP*, pages 1269–1272.

A. Lee, T. Kawahara, and K. Shikano. 2001a. Gaussian mixture selection using context-independent HMM. In *Proc. IEEE ICASSP*, pages 69–72.

A. Lee, T. Kawahara, and K. Shikano. 2001b. Julius — an open source real-time large vocabulary recognition engine. In *Proc. EUROSPEECH*, pages 1691–1694.

T. Matsui, M. Naito, H. Singer, A. Nakamura, and Y. Sagisaka. 1999. Japanese spontaneous speech database with wide regional and age distribution. In *Proc. EUROSPEECH*, volume 5, pages 2251–2254.

T. Morimoto, N. Uratani, T. Takezawa, O. Furuse, Y. Sobashima, H. Iida, A. Nakamura, Y. Sagisaka, N. Higuchi, and Y. Yamazaki. 1994. A speech and language database for speech translation research. In *Proc. ICSLP*, pages 1791–1794.

A. Nakamura, S. Matsunaga, T. Shimizu, M. Tonomura, and Y. Sagisaka. 1996. Japanese speech databases for robust speech recognition. In *Proc. ICSLP*, pages 2199–2202.

S. Young, J. Jansen, J. Odell, D.Ollason, and P. Woodland, 1995. *The HTK Book*.