LIUM and CRIM ASR System Combination for the REPERE Evaluation Campaign

Anthony Rousseau[1], Gilles Boulianne[2], Paul Deléglise[1], Yannick Estève[1],
Vishwa Gupta[2], and Sylvain Meignier[1]

[1] LIUM – University of Le Mans, France
http://www-lium.univ-lemans.fr
[2] Centre de Recherche Informatique de Montréal (CRIM), Québec, Canada
http://www.crim.ca

Abstract. This paper describes the ASR system proposed by the SODA consortium to participate in the ASR task of the French REPERE evaluation campaign. The official test REPERE corpus is composed of TV shows. The entire ASR system was produced by combining two ASR systems built by two members of the consortium. Each ASR system has some specificities: one uses an i-vector-based speaker adaptation of deep neural networks for acoustic modeling, while the other one rescores word-lattices with continuous space language models. The entire ASR system won the REPERE evaluation campaign on the ASR task. On the REPERE test corpus, this composite ASR system reaches a word error rate of 13.5 %.

1 Introduction

REPERE is an evaluation project in the field of people recognition in television documents [2], funded by the DGA (French defence procurement agency) and ending in 2014. Several evaluation tasks were organized, including an evaluation of automatic speech recognition systems on French TV shows.

This paper describes the ASR system proposed by the SODA consortium, including CRIM and LIUM institutions. This system, which combines CRIM's and LIUM's individual ASR systems, won the evaluation task.

Both systems are built on the Kaldi project [14], but each one has some specificities. For instance, CRIM has developed for its system an i-vector-based speaker adaptation of deep neural networks for acoustic modeling [8], while LIUM system has developed a tool to rescore word-lattices by using continuous space language models [15].

In addition to the speaker adaptation approach and the linguistic rescoring of word-lattices, main differences between the two ASR systems are vocabulary, tokenization, training data, and acoustic features. The combination of the two systems provides a very significant reduction of word error rate.

2 ASR System

As seen above, the ASR system which participated in the ASR task of the REPERE evaluation campaign is a composite ASR system. The combination of the two single

P. Sojka et al. (Eds.): TSD 2014, LNAI 8655, pp. 441–448, 2014.
This is preprint prepared by Proceedings editor for Springer International Publishing Switzerland.

ASR systems which are involved in the composite system is made by merging word-lattices. In order to make this merging easier, both ASR systems use the same speech segmentation.

2.1 Speaker Segmentation

To segment the audio recordings and to cluster speech segments by speaker, we used the *LIUM_SpkDiarization* speaker diarization toolkit [12]. This speaker diarization system is composed of an acoustic Bayesian Information Criterion (BIC)-based segmentation followed by a BIC-based hierarchical clustering. Each cluster represents a speaker and is modeled with a full covariance Gaussian. A Viterbi decoding re-segments the signal using GMMs with 8 diagonal components learned by EM-ML, for each cluster. Segmentation, clustering and decoding are performed with 12 MFCC+E, computed with a 10ms frame rate. Gender and bandwidth are detected before transcribing the signal with the two ASR systems.

2.2 LIUM ASR System

The LIUM ASR system built for the REPERE evaluation campaign is based on the Kaldi Speech Recognition Toolkit, which uses finite state transducers (FSTs) for decoding (the general approach is described in [13]). A first step is performed with the Kaldi decoder by using a bigram language model and classical GMM/HMM models to compute a fMLLR matrix transformation. Another step is performed by using the same language model and deep neural network acoustic models. This pass generates word-lattices: an in-house tool, derived from a rescoring tool from the CMU Sphinx project, is used to rescore word-lattices with a 5-gram continuous space language model [15].

In this section we will first present the training data used to estimate LIUM's models, then describe how the system was built using this toolkit.

Training Data The training set used to build LIUM's system consists of 145,781 speech segments from several sources: the radiophonic broadcast ESTER [3] and ESTER2 [4] corpora, which accounts for about 100 hours of speech each; the TV broadcast ETAPE corpus [5], accounting for about 30 hours of speech; the TV broadcast REPERE train corpus, accounting for about 35 hours of speech and other LIUM radio and TV broadcast data for about 300 hours of speech, which have been segmented using the speaker diarization system described above. The training dictionary has 107.603 phonetized entries. Table 1 summarizes the characteristics of each dataset.

For language modeling, the training data is composed of the manual transcriptions from the training corpus used to estimate the acoustic models, of articles extracted from of TV websites, of articles extracted from Google News, of the French Gigaword corpus, of articles from newspaper 'Le Monde'. All of these data were collected before January 2013.

Table 2 presents of the number of words in each corpus in the training corpus used to estimate language models.

Table 1. Characteristics of the training data for acoustic modeling.

Sources	Speech	Segments
ESTER	100h	12,902
ESTER2	100h	15,162
ETAPE	30h	8,378
REPERE	35h	10,269
LIUM v8	300h	99,070
Total	565h	145,781

Table 2. Characteristics of the training data for language models.

Sources	Number of words
Manual transcriptions from the training corpora used to train the acoustic models	8M
Articles from TV websites (≤2012)	5M
Google News (≤2012)	204M
French Gigaword (≤2012)	1015M
Newspapers (≤2012)	366M
Subtitles of TV Newspaper (≤2012)	11M
Total	1609M

Acoustic Modeling The GMM-HMM (Gaussian Mixture Model – Hidden Markov Model) models are trained on 13-dimension PLP features with first and second derivatives by frame. By concatenating the four previous frames and the four next frames, this corresponds to $39 \times 9 = 351$ features projected to 40 dimensions with linear discriminant analysis (LDA) and maximum likelihood linear transform (MLLT). Speaker adaptive training (SAT) is performed using feature-space maximum likelihood linear regression (fMLLR) transforms. Using these features, the models are trained on the full 565 hours set, with 12,000 tied triphone states and 450,000 Gaussians. On top of these models, we train a deep neural network (DNN) based on the same fMLLR transforms as the GMM-HMM models and on state-level minimum Bayes risk (sMBR) [10] as discriminative criterion. Again we use the full 565 hours set as the training material. The resulting network is composed of 7 layers for a total of 42.5 millions parameters and each of the 6 hidden layers has 2,048 neurons. The output dimension is 9,866 units and the input dimension is 440, which corresponds to an 11 frames window with 40 LDA parameters each.

Weights for the network are initialized using 6 restricted Boltzmann machines (RBMs) stacked as a deep belief network (DBN). The first RBM (Gaussian-Bernoulli) is trained with a learning rate of 0.01 and the 5 following RBMs (Bernoulli-Bernoulli) are trained with a rate of 0.4. The learning rate for the DNN training is 0.00001. The segments and frames are processed randomly during the network training with stochastic gradient descent in order to minimize cross-entropy between the training data and network output. When these training steps are done, the last step of training is processed, by applying the minimum Bayes risk criterion, as indicated above. To speed up the learning process, we used a general-purpose graphics processing unit (GPGPU) and the CUDA toolkit for computations.

Language Modeling The vocabulary used in the LIUM ASR system has 160K words. The bigram language model used during the decoding with Kaldi is trained on the data presented in section 1.1 by using the SRILM toolkit [16]. No cut-off was applied and the modified Kneser-Ney discounting is applied.

To rescore word-lattices generated by Kaldi, trigram and quadgram LMs are trained with the same toolkit. A 5G continuous-space language model (CSLM) is also estimated for the final lattice rescoring. No cut-off is applied and the same discounting method as for the bigram language model is applied.

2.3 CRIM ASR System

This system is also based on the Kaldi toolkit, with the addition of DNN speaker adaptation based on i-vectors [8].

Training data CRIM training data to estimate acoustic models contains the same ESTER, ESTER 2, ETAPE and REPERE corpora as LIUM's, for a total of 335 hours of audio: this number is higher than the number of hours used by LIUM from these corpora because LIUM put aside about 50 hours from ESTER 2.

In addition to these 335 hours, CRIM had 178 hours of internally transcribed audio from French TV broadcasts in Quebec. Overall, CRIM had 513 hours of transcribed audio for training. In all the training audio, speaker segments were manually labeled in order to facilitate speaker-adapted training.

Acoustic Modeling For training the deep neural network (DNN) using back propagation, 3 hours of the training audio were set aside for validation. CRIM uses TRAP (TempoRAl Pattern) features [6] extracted from filter-bank as input to the neural net. To compute TRAP features, 23-dimensional filterbank features are normalized to zero mean per speaker. Then 31 frames of these 23-dimensional filterbank features (15 frames on each side of current frame) are spliced together to form a 713-dimensional feature vector. This 713-dimensional feature vector is transformed using a Hamming window (to emphasize the center), passed through a discrete cosine transform and the dimensionality is reduced to 368. This 368-dimensional feature vector is globally normalized to have zero mean and unit variance.

The i-vector extractor is trained from the same data used for training the DNN, using speaker labels from the transcriptions. At test time, for each speaker identified by the automatic segmentation, one i-vector of dimension 100 is extracted. The TRAP features are then augmented with the 100-dimensional i-vector corresponding to the current speaker. This 468-dimensional feature vector is then input to the 7-layer DNN, as illustrated in Figure 1. The feature vector is advanced by one frame every time (note that the i-vector part stays fixed for a given speaker).

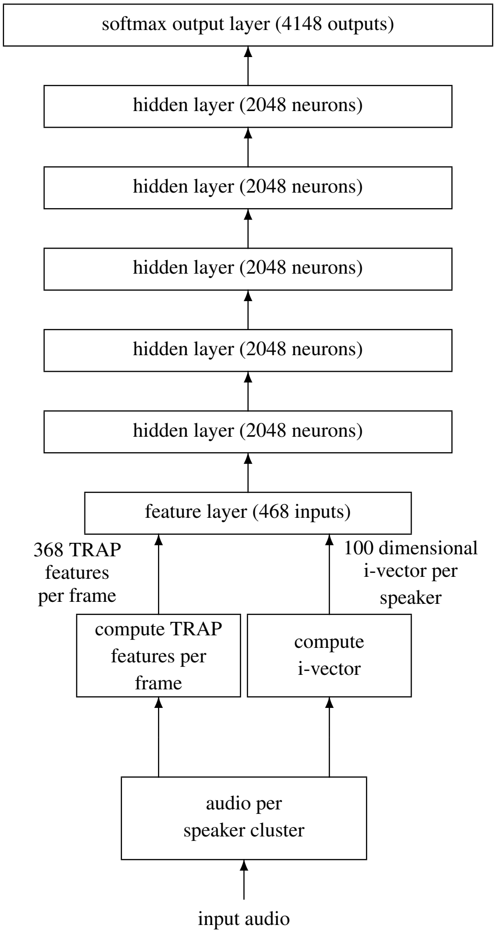
softmax output layer (4148 outputs)
hidden layer (2048 neurons)
hidden layer (2048 neurons)
hidden layer (2048 neurons)
hidden layer (2048 neurons)
hidden layer (2048 neurons)
feature layer (468 inputs)
368 TRAP features per frame
100 dimensional i-vector per speaker
compute TRAP features per frame
compute i-vector
audio per speaker cluster
input audio

Fig. 1. Deep neural network architecture used for speaker adaptation of acoustic models in the CRIM ASR system.

Language Modeling Sources for CRIM include broadcast news transcriptions from EPAC and ESTER campaigns, transcripts from ETAPE, 350,000 sentences selected from French Gigaword database, and Google 4-grams (closely following [7]). Entropy-based pruning was applied to reduce language model size to 1.8M trigrams for search and 20M quadgrams for rescoring word lattices. Perplexities on the REPERE development text are 162 for the search trigram and 134 for the rescoring quadgram, with an out-of-vocabulary rate of 0.65%. The initial vocabulary was selected by taking words with the highest frequency count weighted inversely with source size until a vocabulary size of 100,000 words was obtained. To this, words from REPERE training transcripts were added, as well as proper names found in ETAPE, EPAC, ESTER sources, and also French departments, Paris metro stations, and French acronyms taken from the Web. The final vocabulary was 144,000 words.

2.4 Word-lattice Merging

CRIM and LIUM used the same audio segmentation, provided by the *LIUM_SpkDiarization* speaker diarization system. Using the same segmentation makes easier the merging between the two ASR outputs: final outputs were obtained by merging word-lattices provided by both ASR systems.

Both LIUM and CRIM ASR systems provide classical word-lattices with usual information: words, temporal information, acoustic and linguistic scores. Before merging lattices, for each edge, these scores are replaced by its *a posteriori* probability. Posteriors are computed for each lattice independently, then weighted by $\frac{1}{n}$, where n is the number of word-lattices to be merged (here, $n = 2$). In our experiments, we did not find significant improvements by using more tuned weights.

For each speech segment, the use of weighted posteriors allows to merge starting (respectively ending) nodes from LIUM and CRIM lattices together into a single lattice in order to process directly with an optimized version of the consensus network confusion algorithm [11]. This optimization reduces very significantly the computation time by managing temporal information during the clustering steps.

3 Experimental Results

Experimental Data This study was conducted on two corpora from the REPERE French evaluation campaign [9]. The development corpus (dev) is composed of 28 TV shows. This corpus corresponds to the test corpus of the first evaluation which took place in January 2013. The test corpus (test) is composed of 62 TV shows. It corresponds to the test set of the second evaluation (January 2014). Shows are recorded from the two digital French terrestrial television stations BFM and LCP.

These corpora are balanced between prepared speech, with 23 broadcast news, and more spontaneous speech, from 67 political discussions or street interviews. Only a part of the recordings are annotated, giving respectively a total duration of 3 hours for dev corpus and 10 hours for the test corpus.

Results and Discussion A first evaluation on the ASR task was organized last year in 2013, in which the LIUM ASR system ranked first, on similar but different test data. This system appears in Table 3 under the name *old 2013 LIUM* system: it can be used to measure improvements achieved since last year.

The *old 2013 LIUM* ASR system was based on the CMU Sphinx toolkit, with some improvements, for instance the use of hybrid MLP/HMM acoustic models. A variant of this system is described in [1].

The main difference between the new ASR system developed by LIUM and the old one comes from the use of DNN acoustic models and the use of the *finite state machine* paradigm. These functionalities are both offered by the Kaldi toolkit. Notice that the linguistic rescoring tool is the same one in both LIUM ASR systems. With the same language models and the same training data for the estimation of acoustic models, the word error rate (WER) of the LIUM ASR system is reduced of 2.6 points (14%) to 16.0%.

The CRIM system achieves a word error rate of 16.3%. When the linguistic rescoring tool of LIUM system is applied to CRIM word-lattices (called CRIM+CSLM in Table 3), the WER is 1 point smaller than the WER of the LIUM system. This can be explained by better acoustic models provided by the DNN adaptation approach proposed by CRIM.

Combining the single-best hypothesis of each system with ROVER (and by using confidence measures) fails to provide an improvement (line ROVER in Table 3).

In contrast, merging word lattices achieves a large reduction in error over both individual systems (line CRIM $\oplus$ LIUM in Table 3), bringing the WER down by about 2 points (13.1% relative) when applied to LIUM and initial CRIM systems.

Notice that when applied to LIUM and CRIM+CSLM, the WER is reduced by 1.5 point (10% relative). The same training data were used to train the CSLMs of the CRIM+CSLM and the LIUM systems: this may explain this smaller improvement provided by the merging process.

Table 3. Word error rates on REPERE test corpus (TV shows)

ASR system	WER
old 2013 LIUM	18.6%
LIUM	16.0%
CRIM	16.3%
ROVER(CRIM,LIUM)	16.3%
CRIM $\oplus$ LIUM	13.9%
CRIM+CSLM	15.0%
CRIM+CSLM $\oplus$ LIUM	13.5%

4 Conclusion

Both LIUM and CRIM ASR systems are based on the Kaldi toolkit. Each one has noticeable specificities: the CRIM system uses a DNN speaker adaptation approach, while the LIUM system uses a 5g CSLM to rescore word-lattices. The word-lattice merging used in this work in order to build a composite ASR system permits to get significant improvements in terms of word error rate, and is very simple to use: no constraint about vocabulary, tokenization, nature of acoustic models. Only classical word-lattices are necessary, with acoustic and linguistic scores in order to compute posteriors. Merging LIUM and CRIM ASR systems was easy, and these systems were sufficiently accurate and complementary to get such performances.

The *old 2013 LIUM* ASR system, which won the two last evaluation campaigns on French language in 2012 and 2013, achieves a WER of 18.6% on the test data of the 2014 REPERE campaign. From this starting point, the composite system presented in this paper reduces the WER down to 13.5% on the test set (WER reduction of 27% relative), a significant advance in state-of-the-art French ASR.

References

1. Bougares, F., Deléglise, P., Esteve, Y., Rouvier, M.: LIUM ASR system for Etape French evaluation campaign: experiments on system combination using open-source recognizers. In: Text, Speech, and Dialogue. pp. 319–326. Springer (2013)
2. Galibert, O., Kahn, J.: The first official REPERE evaluation. In: First Workshop on Speech, Language and Audio in Multimedia (SLAM). pp. 43–48. Marseille, France (2013)
3. Galliano, S., Geoffrois, E., Gravier, G., f. Bonastre, J., Mostefa, D., Choukri, K.: Corpus description of the Ester evaluation campaign for the rich transcription of French broadcast news. In: 5th international Conference on Language Resources and Evaluation (LREC). pp. 315–320 (2006)
4. Galliano, S., Gravier, G., Chaubard, L.: The Ester 2 evaluation campaign for the rich transcription of french radio broadcasts. In: Interspeech (2009)
5. Gravier, G., Adda, G., Paulsson, N., Carré, M., Giraudel, A., Galibert, O.: The ETAPE corpus for the evaluation of speech-based TV content processing in the French language. In: Eighth International Conference on Language Resources and Evaluation (LREC). pp. 114–118. Istanbul, Turkey (2012)
6. Grézl, F.: TRAP-based Probabilistic Features for Automatic Speech Recognition. Ph.D. thesis, dept. Computer Graphics & Multimedia, Brno University of Technology (2007)
7. Gupta, V., Boulianne, G., Osterrath, F., Ouellet, P.: CRIM's french speech transcription system for ETAPE 2011. In: WOSSPA (2013)
8. Gupta, V., Kenny, P., Ouellet, P., Stafylakis, T.: I-vector-based speaker adaptation of deep neural networks for french broadcast audio transcription. In: ICASSP. Florence, Italy (2014)
9. Kahn, J., Galibert, O., Quintard, L., Carre, M., Giraudel, A., Joly, P.: A presentation of the REPERE challenge. In: International Workshop on Content-Based Multimedia Indexing (CBMI). pp. 1–6 (2012)
10. Kingsbury, B.: Lattice-based optimization of sequence classification criteria for neural-network acoustic modeling. In: ICASSP. pp. 3761–3764 (2009)
11. Mangu, L., Brill, E., Stolcke, A.: Finding consensus in speech recognition: word error minimization and other applications of confusion networks. Computer Speech & Language 14(4), pp. 373–400 (2000)
12. Meignier, S., Merlin, T.: LIUM SpkDiarization: an open source toolkit for diarization. In: CMU SPUD Workshop. Dallas, Texas, USA (2010)
13. Mohri, M., Pereira, F., Riley, M.: Speech recognition with weighted finite-state transducers. In: Springer Handbook of Speech Processing, pp. 559–584. Springer Berlin Heidelberg (2008)
14. Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlíček, P., Qian, Y., Schwarz, P., Silovský, J., Stemmer, G., Veselý, K.: The Kaldi Speech Recognition Toolkit. In: ASRU Workshop, pp. 1–4 (2011)
15. Schwenk, H.: CSLM – a modular open-source continuous space language modeling toolkit. In: Interspeech, pp. 1198–1202. Lyon, France (2013)
16. Stolcke, A.: SRILM – an extensible language modeling toolkit. In: Interspeech. pp. 901–904 (2002)