IberSPEECH 2021
24-25 March 2021, Valladolid, Spain

The Biometric Vox System for the Albayzin-RTVE 2020 Speech-to-Text Challenge

Roberto Font[1], *Teresa Grau*[1]

[1]Biometric Vox S.L.

roberto.font@biometricvox.com, teresa.grau@biometricvox.com

Abstract

This paper describes the system developed by Biometric Vox for the Albayzin Speech-To-Text Challenge organized as part of the Iberspeech 2020 conference. The system uses speaker diarization to segment the audio into speaker-homogeneous segments and uses this information to compute speaker-dependent fMLLR transformed features. These speaker-adapted features are the input to a DNN acoustic model which is trained for the domain at hand using a semi-supervised self-training procedure. Finally, a RNN language model is used for lattice rescoring and producing the final transcription. Our system achieves 22% WER on the test portion of the RTVE2018 database and 30,26% on the 2020 evaluation set.

Index Terms: speech recognition, Hybrid DNN-HMM, semi-supervised training, self-training

1. Introduction

The aim of this paper is to provide a complete description of the system developed by Biometric Vox for the Albayzin-RTVE 2020 Speech-to-Text Challenge. The task for this challenge is the automatic transcription of TV shows that belong to diverse genres and broadcast by the public Spanish Television (RTVE), covering a wide range of realistic and challenging conditions: spontaneous and read speech, different accents, background noise, songs, laughs, yells, quick short-turn conversations...

Our main contribution is the use of a very simple yet effective semi-supervised learning method for acoustic model training. Starting from a commercial off-the-shelf system that was developed for the automatic transcription of town hall plenaries, we use self-training to adapt this system to the domain at hand without any human intervention. The transcriptions produced by this initial system are used to train a new system and this procedure can be repeated iteratively. For this work, we performed two of these self-training iterations.

The rest of the paper is organized as follows: in Section 2 the RTVE database and the data used to train the acoustic model and language models are described. Section 3 presents the ASR system and describes its components and characteristics. Results are shown in Section 4, and, finally, we summarize our results and conclusions in Section 5.

2. Data Resources

2.1. RTVE database

The RTVE2018 database [1] has a total of 569 hours and 22 minutes of audio extracted from 17 different TV shows broadcast by RTVE (Radio Televisión Española) from 2015 to 2018. Most shows are related to news, debates, social gatherings and documentaries. The database is divided into 4 subsets: a training set, 2 development sets, *dev1* and *dev2*, and a test set, as summarized in Table 1.

Table 1: *Composition of the different RTVE subsets*

Subset	# hours	# different shows
RTVE2018		
train	500h	16
dev1	52:31:51	5
dev2	15:09:25	2
test	39:07:15	8
RTVE2020		
dev	03:55:31	2
test	67:23:29	15

It is worth noting that the *dev1*, *dev2* and *test* portions have human-revised transcriptions while, in the case of the *train* set, it contains subtitles that were generated through a re-speaking procedure resulting in a no-verbatim word transcription.

Additionally, the database includes a text corpora extracted from all the subtitles broadcast by the RTVE 24H Channel during 2017. The subtitles contain approximately 56M words.

The RTVE2020 database [2] is also a collection of TV shows from the public Spanish Television (RTVE) that were broadcast from 2018 to 2019. The database is composed of 70 hours and 18 minutes of audio belonging to 15 different shows. Only two shows are in common with the 2018 dataset: *Comando Actualidad* and *Millenium*. For the challenge, no transcriptions or subtitles were provided for this set. The database is divided into 2 subsets: *dev* and *test*. The latter one is the challenge evaluation set. Shows' genre for this set is related to entertainment: series, sketches, reports, entertainment shows...

TV show's genre and format are quite different between the two sets; each set covers a different range of conditions. For the RTVE2018 database: spontaneous and read speech, long-turn conversations, different accents and background noise. And for RTVE2020: songs, laughs, yells, quick short-turn conversations and colloquial vocabulary.

For system development, we used the *dev1* portion of the RTVE2018 database as our internal development dataset and the RTVE2018 *test* portion as our test set.

2.2. Hybrid (DNN-HMM) acoustic model

As already discussed, the training portion of the RTVE2018 database contains imprecise transcriptions, since the subtitles have been generated by a re-speaking procedure resulting in a

[1]http://catedrartve.unizar.es/reto2018/RTVE2018DB.pdf

[2]http://catedrartve.unizar.es/reto2020/RTVE2020DB.pdf

10.21437/IberSPEECH.2021-21

no verbatim word transcription. This prevents the usage of this material in a standard acoustic model training setup.

To account for this difficulty, we have used a semi-supervised learning method that was developed for our commercial automatic transcription system *Transcribe Vox* [3]. This system, which automatically transcribes town hall plenaries and similar meetings, was trained using an iterative self-training procedure. First, an initial system was bootstrapped from a small amount of labelled data, then this initial system was used to produce transcriptions on a large amount of unlabelled data. These transcriptions were used to train a new system and this procedure was repeated iteratively to produce increasingly accurate transcriptions.

For the Albayzin-RTVE 2020 Speech-to-Text Challenge we followed the same approach using this pre-existing model to generate transcriptions for the training portion of the RTVE2018 database. These transcriptions were used, instead of the provided subtitles, to train our system.

To sum up, the acoustic model is trained on:

- A set of roughly 530 hours of town hall meetings and the transcriptions generated by the initial system. This material, publicly available, was downloaded from the Internet trough the different platforms offered to citizens to review the meetings.
- RTVE2018 training set and the transcriptions generated by the initial system (trained on the above set).

To increase model robustness, we applied data augmentation for DNN training. We produced volume and speed perturbations and added reverberation and noise using the Musan corpus [1]. A subset of the augmented data was selected obtaining a final training set of approximately $1,800$ hours.

2.3. Language modelling

To train the LMs the following text corpora were used:

In-domain data: The RTVE subtitles provided in the RTVE2018 database. The subtitles contain approximately 56M words.

Out-of-domain data: The combination of the Spanish portion of the Europarl corpus [2] and the automatically-generated transcriptions of the town hall meetings described in the previous section, with a total of around 60M words.

The text was preprocessed by the usual procedure of normalizing, removing punctuation, expanding the most usual contractions and transliterating numbers. We used a vocabulary of 101K words including the 320 most frequent words in the RTVE subtitles that were not present in the off-the-shelf lexicon.

Two kinds of LMs have been trained:

- **3-gram LM:** A trigram language model obtained by linear interpolation of two models: one trained on in-domain data, and the other on the out-of-domain dataset. The optimal interpolation weight was tuned on the development set.
- **RNNLM:** An RNN language model trained on the combination of both datasets.

[3] https://biometricvox.com/transcribevox/

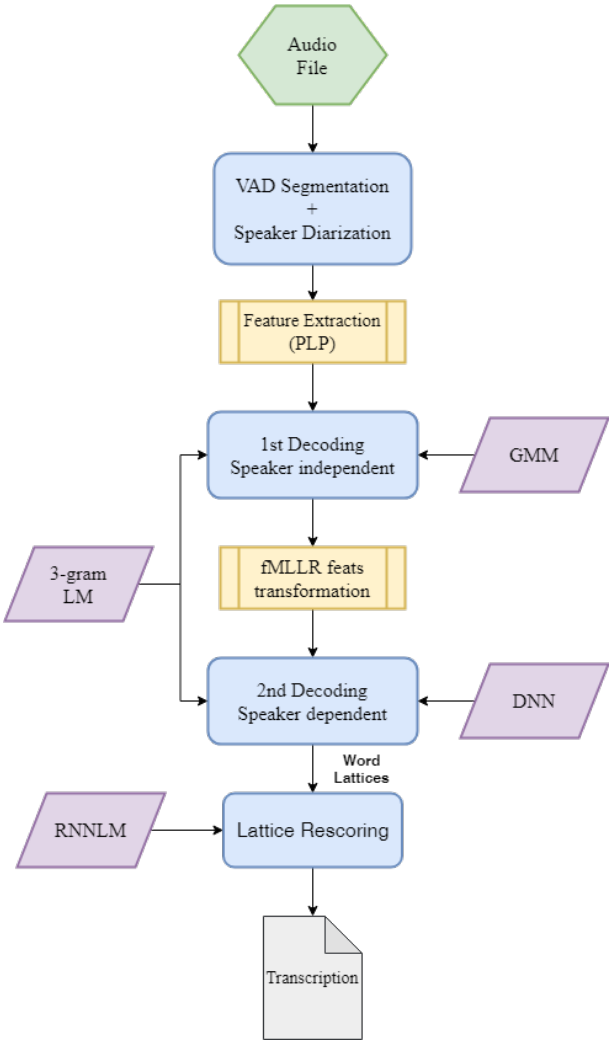

Figure 1: *System flowchart*

3. System Components

The transcription process is composed of the following steps, which were implemented using the Kaldi [3] toolkit:

- Initial segmentation and silence removal using Kaldi standard energy-based VAD.
- Speaker diarization to merge together the segments belonging to the same speaker.
- First-pass speaker independent decoding using a triphone GMM system. This first-pass decoding is used to compute fMLLR transforms.
- Second-pass decoding using fMLLR-transformed features and a DNN model.
- Lattice rescoring [4] using an RNN language model.

Figure 1 shows an schematic view of this process.

3.1. Speaker diarization

To merge together all segments belonging to the same speaker and compute speaker-adapted features, we perform a speaker

diarization step using the system developed for the Albayzin-RTVE 2020 Speaker Diarization and Identity Assignment Challenge. The system is based on the DNN x–vector paradigm [5] and consists of the following steps:

- Acoustic feature extraction (23 MFCC features) and Voice Activity Detection using an energy-based VAD.
- X–vector embedding extraction.
- Embedding post-processing (length-normalization, centering, whitening, LDA)
- PLDA scoring.
- Agglomerative Hierarchical Clustering.

The embedding extractor is trained on NIST SRE 04-10, MIXER6, Switchboard and VoxcelebCat. VoxcelebCat is the result of concatenating all excerpts from the same video into one longer file and combining Voxceleb 1 train, Voxceleb 2 dev and Voxceleb 2 test. The samples from NIST SRE 04-10, MIXER6 and Switchboard were upsampled to 16kHz. As is usual, we augment the training data by generating perturbed versions using Musan [1] noise and reverbation.

For the embedding extractor, we used a baseline TDNN x-vector architecture as in the Kaldi SRE 16 recipe (Table 2).

Table 2: *Embedding extraction architecture for speaker diarization*

Layer type	Layer context	Size
TDNN-ReLU-batchnorm	t-2:t+2	512
TDNN-ReLU-batchnorm	t-2, t, t+2	512
TDNN-ReLU-batchnorm	t-3, t, t+3	512
ReLU-batchnorm	t	512
ReLU-batchnorm	t	1500
Stats Pooling (mean+stddev)	T	2x1500
ReLU-batchnorm		512
ReLU-batchnorm		512
Softmax		# speakers

Embeddings are extracted using a sliding window of 3 seconds and a second and a half hop. Then, they are length-normalized, projected from 512 dimensions to 150 using LDA and scored using PLDA. Finally, segments are clustered using Agglomerative Hierarchical Clustering (AHC).

Both LDA and PLDA are trained on VoxcelebCat.

3.2. First-pass GMM acoustic model

The GMM model used in the first-pass speaker independent decoding is an LDA+MLLT+SAT triphone system with $4,360$ classes (tied-state triphones) and $60,094$ gaussians that has as input 13-dimensional PLP features spliced across 7 consecutive frames and projected to 40 dimensions trough the LDA+MLLT transformation. The training set for this model is the same described in Section 2.2 except for the augmented data, that was not included. As already mentioned, this first-pass decoding is used to compute the fMLLR-transformed features, the input for the DNN acoustic model.

3.3. DNN acoustic model

The DNN acoustic model architecture is based on Kaldi's *chain* model [6]. A factorized time-delay neural network (TDNN-F) [7], which is structurally the same as a TDNN [8], but is trained from a random start with one of the two factors of each matrix constrained to be semi-orthogonal.

Our network consists of 7 tdnn-f layers with dimension $1,536$ and linear bottleneck layers of dimension 256, ReLU activation function and batch normalization. For the loss function we used L2 regularization to prevent overfitting. Table 3 shows the architecture of the tdnnf. Instead of using ivectors for speaker adaptation, as is often the case, we use fMLLR-transformed features as input for the DNN. These transformations are computed using the first-pass decoder described above. To improve the accuracy of this speaker adaptation, we do an initial speaker diarization step to try and group together all segments belonging to the same speaker.

Table 3: *Factorized TDNN architecture. Note that Batch-Norm applied after each ReLU is omitted for brevity.*

Layer type	Context factor1	Context factor2	Size	Inner size
tdnn-ReLU	t		1536	
tdnnf-ReLU	t-1, t	t, t+1	1536	160
tdnnf-ReLU	t-1, t	t, t+1	1536	160
tdnnf-ReLU	t-1, t	t, t+1	1536	160
tdnnf-ReLU	t	t	1536	160
tdnnf-ReLU	t-3, t	t, t+3	1536	160
tdnnf-ReLU	t-3, t	t, t+3	1536	160
tdnnf-ReLU	t-3, t	t, t+3	1536	160
Linear			256	
Dense-ReLU-Linear			256	1536
Dense			#Targets	

TDNN training time for 4 epochs and 1763 iterations was 40 hours 52 minutes on an NVIDIA Geforce GTX 1080 GPU.

3.4. Language models

As mentioned in Section 2.3, we used a 3-gram LM which is the linear interpolation of in-domain and out-of-domain LMs for decoding and a Recurrent Neural Network LM for the final lattice rescoring.

For the 3-gram LM, the models were trained using the SRI Language Modeling Toolkit [9], and the optimal interpolation weight was tuned on the development set (*dev1*). Table 4 shows the perplexities for the out-of-domain, in-domain and interpolated LMs.

Table 4: *Language Model perplexity*

LM	Perplexity
Out-of-domain	440.65
In-domain	235.29
Interpolation	205.36

The RNNLM was also trained using the Kaldi toolkit. The network is trained with an architecture of 5 hidden layers, each with 800 neurons, where TDNN layers with ReLU activation function, and LSTM layers are combined. Figure 2 shows the topology of the network used.

The RNNLM was trained for 480 iterations, being the 478^{th} the best iteration. 10K sentences from the train set were used as development set.

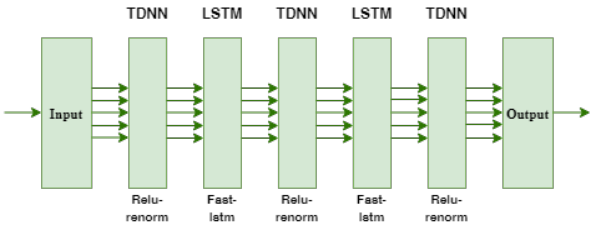

Figure 2: *RNNLM topology*

4. Results

The Table 5 shows the results of our system on the development and test portions of the RTVE2018 database.

Three systems are evaluated. The first one is the off-the-shelf commercial system described in Section 2.2. The other two systems are the result of the first and second self-training iterations as described in Section 2.2. No more self-training iterations were made due to time constraints. "RTVE2.0" system was our primary submission for the challenge. The evaluation results provided by the organizers of our primary system for the 2020 S2T Challenge's evaluation set are shown in Table 6. The table shows the total number of words and the average Word Error Rate (WER) by TV show, and also the global average WER.

We can see that the best result is obtained for the show *Los desayunos de TVE* (LD), which is related to news content and long-turn conversation almost without interruptions between speakers. On the other hand, the highest WER is obtained for the shows *Como nos Reíamos* (CN) and *Si Fueras Tú* (SFT), which are entertainment shows with sketches with a loud voice tone and loud background music.

The decoding was performed on an Intel(R) Core(TM) i7-5820K CPU @ 3,30GHz with 6 cores, and the processing time for the *dev2* subset was 4 hours and 27 minutes.

Table 5: *WER (%) on the RTVE2018 dataset*

System	dev2	Test
Off-the-shelf system	21.8%	-
RTVE1.0 + RNNLM_RTVE	20.3%	23.6%
RTVE2.0 + RNNLM_RTVE	17.8%	22.0%

5. Conclusions

We have presented our ASR system submitted to the Albayzin Speech-To-Text Challenge. This challenge, which focuses on the automatic transcription of TV shows, provides researchers with an ASR task with some very interesting and challenging features. Indeed, the provided training data contains imprecise transcriptions making it difficult to use it in a standard acoustic model training setup. In addition, the TV shows include some of the most challenging conditions for any speech recognition system: spontaneous speech, different accents, noisy backgrounds, overlapped speakers... Furthermore, those conditions are different between the RTVE2018 and RTVE2020 database. The first one focuses mainly in news content, where the speech tends to be slow and with long-turn conversations. Meanwhile, RTVE2020 focuses on entertainment shows with songs, sketches with a loud voice tone, quick short-turn conversations...

Table 6: *Evaluation results RTVE2020. #W column is for the total number of words and %WER for the average Word Error Rate by TV show.*

TV Show	#W	(%)WER
AT	108771	26.25
BN	6224	55.16
BR	6139	50.68
CA	37942	36.27
CN	20628	69.20
EP	21402	34.58
IM	24474	49.04
LD	121067	13.86
MC	85464	35.23
ML	16949	23.33
NFMY	1433	45.57
SFT	2108	54.87
VC	37458	36.32
VE	26314	24.29
WU	3406	52.17
Global	519779	30.26

Our main contribution is the use of a semi-supervised self-learning method to train the system without the need of labelled data for the domain at hand. The initial system, trained on data from town hall plenary sessions, was refined iteratively using the RTVE training data without the need of any transcriptions. The town hall plenaries used as a starting point are characterized by structured discourses with long speaking turns. Unsurprisingly, the system performed better on RTVE2018 data, which is closer to that style. However, the proposed approach proved to be effective, adapting to the new domain and providing better results at each iteration. If we focus, for example, on the TV show *Comando Actualidad* (CA), which is present in both datasets, we can see that the %WER obtained with the off-the-shelf system was 72.6%, 52.9% for the first self-training iteration and 36.27% for the last one. This suggest that the proposed strategy could be used to successfully adapt to new challenging domains, like the 2020 evaluation set, by leveraging on-domain unlabelled data and performing more self-training iterations.

6. Acknowledgements

The authors would like to thank the organizers of the Albayzin Challenge.

7. References

[1] D. Snyder, G. Chen, and D. Povey, "MUSAN: A music, speech, and noise corpus," 2015.

[2] P. Koehn, "Europarl: A parallel corpus for statistical machine translation," 2005.

[3] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The kaldi speech recognition toolkit," in *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*. IEEE Signal Processing Society, Dec. 2011, iEEE Catalog No.: CFP11SRW-USB.

[4] H. Xu, T. Chen, D. Gao, Y. Wang, K. Li, N. Goel, Y. Carmiel, D. Povey, and S. Khudanpur, "A pruned rnnlm lattice-rescoring algorithm for automatic speech recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 5929–5933.

[5] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," 04 2018, pp. 5329–5333.

[6] D. Povey, V. Peddinti, D. Galvez, P. Ghahremani, V. Manohar, X. Na, Y. Wang, and S. Khudanpur, "Purely sequence-trained neural networks for asr based on lattice-free mmi," 09 2016, pp. 2751–2755.

[7] D. Povey, G. Cheng, Y. Wang, K. Li, H. Xu, M. Yarmohammadi, and S. Khudanpur, "Semi-orthogonal low-rank matrix factorization for deep neural networks," 09 2018, pp. 3743–3747.

[8] V. Peddinti, D. Povey, and S. Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts," in *INTERSPEECH 2015 - 15th Annual Conference of the International Speech Communication Association*, September 2015.

[9] A. Stolcke, "Srilm — an extensible language modeling toolkit," *Proceedings of the 7th International Conference on Spoken Language Processing (ICSLP 2002)*, vol. 2, 07 2004.