IberSPEECH 2018
21-23 November 2018, Barcelona, Spain

Building an Open Source Automatic Speech Recognition System for Catalan

Baybars Külebi[1], *Alp Öktem*[2,1]

[1]Col·lectivaT SCCL, Barcelona, Spain
[2]Universitat Pompeu Fabra, Barcelona, Spain
bkulebi@collectivat.cat, alp@collectivat.cat

Abstract

Catalan is recognized as the largest stateless language in Europe hence it is a language well studied in the field of speech, and there exists various solutions for Automatic Speech Recognition (ASR) with large vocabulary. However, unlike many of the official languages of Europe, it neither has an open acoustic corpus sufficiently large for training ASR models, nor openly accessible acoustic models for local task execution and personal use. In order to provide the necessary tools and expertise for the resource limited languages, in this work we discuss the development of a large speech corpus of broadcast media and building of an Catalan ASR system using CMU Sphinx. The resulting models have a WER of 35,2% on a 4 hour test set of similar recordings and a 31.95% on an external 4 hour multi-speaker test set. This rate is further decreased to 11.68% with a task specific language model. 240 hours of broadcast speech data and the resulting models are distributed openly for use.

Index Terms: speech recognition, audio corpus, Catalan, open source

1. Introduction

Development of technology that involves spoken input necessitates three components in order to accomplish the conversion of human speech to machine readable form: (1) Automatic speech recognition (ASR) system, (2) acoustic model and (3) language model. There exists various open source alternatives for ASR systems such as *CMU Sphinx*, *Kaldi* and *DeepSpeech*, however, publicly available acoustic and language models to use with these applications are mainly available for highly-resourced languages. As speech corpora development needed to build the acoustic models is a high-cost process, it is harder to find open source models for lesser-used languages. This puts these languages in major disadvantage in terms of accessibility in technologies with a voice interface.

Catalan with an estimated 9.1 million speakers among 4 different countries[1], is the ninth most spoken language in Europe and second in Spain. There has been development in Catalan speech technology both from inside and outside of Catalonia, and both research and commercial oriented. Research projects that involve Catalan speech recognition include large-vocabulary *TECNOPARLA* [1] and an earlier telephone conversation targeted project [2], both from Universitat Politécnica de Catalunya (UPC). The speech recognition models resulting from these projects are not available for public use. Only the corpus used to develop the former one is available on demand and is of limited size. On the commercial spectrum, cloud ASR services by companies like Google, Facebook and Speechmatics provide services in Catalan. However, these services are only provided through a centralized server with a fee or a usage limit and does not guarantee data privacy. Also, the fact that they are closed-source clears away any possibility for customization to different needs. Regarding these, although there seems to be reasonable research and development in Catalan ASR systems, there exists no project with an emphasis on free and open sourced access to its resources.

In this work, we present our development of an open source large vocabulary ASR system for Catalan. To build a free and open system, we incorporated a variety of free tools and resources for our use. We chose to use the HMM-based speech recognition system CMU Sphinx principally for its practicality in application development. Although state-of-the-art systems are neural network based, CMU Sphinx is still a popular choice as an ASR toolkit for its computationally low-cost architecture, active community of developers and rich documentation. As for training data we exploited the broadcast media material publicly available through the Catalan public television *TV3*. Both acoustic and text resources were collected automatically and processed for model training.

With the motivation of making the language accessible to speech technology developers, we have made both the models and the dataset available online. The ready-to-use models, instructions for deployment and a basic script for decoding are distributed in https://github.com/collectivat/cmusphinx-models. 240 hours of broadcast speech data collected during this process is also accessible through this repository.

2. Speech Recognition System

Basic architecture of an ASR system consists of five elements as can be seen in the Figure 1. Before the central decoder can accept speech signals, they need to be converted into a sequence of fixed size acoustic vectors through a process called *feature extraction*. Later the recognizer makes use of the acoustic model, the language model and the pronunciation dictionary in order to decode the vector sequences into the most likely word sequences they represent. In most cases, a language model consists of N-grams in which the probability of occurrence of a word depends on its $N-1$ predecessors and the pronunciation dictionary provides the mapping between each word and its phonetically written form.

The core part of the recognition relies on how the speech is modeled, which depends on how acoustic model parameters are defined and furthermore how they are trained. In many classic ASR systems, Hidden Markov Models (HMMs) are used for modeling the phones (or tri-phones) as finite states each with associated probability distributions. Within the ASR architecture, HMMs are used for evaluating the probability of given speech features (*forward-backward algorithm*). Furthermore, their use permit the estimation of the the best model parameters for pro-

[1]According to the estimate by Secretary of Language Policies of Government of Catalonia

10.21437/IberSPEECH.2018-6

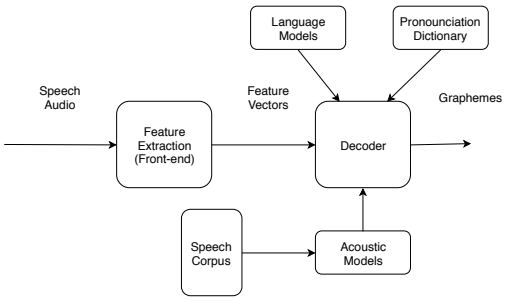

Figure 1: *General architecture of an ASR system.*

ducing the given speech features (*Baum-Welch algorithm*) as well as decoding the speech feature observables to the most likely state sequences (*beam search algorithm*). For a review of use HMM-based ASR see [3].

For our task at hand we decided to use the CMU Sphinx ASR system, which has been developed at Carnegie Mellon University over many decades starting from the original Sphinx-I [4] to its more recent and advanced incarnations of Sphinx-3 [5] and Sphinx-4 [6]. For this work, we specifically used the PocketSphinx package within the Sphinx-5prealpha which incorporates algorithms and codes from the previous CMU Sphinx packages [7, 8, 9]. Comparison studies of open-source ASR tools based on training of similar datasets show that Kaldi is far superior in terms of recognition accuracy, mostly due to its inclusion of advanced techniques such as Deep Neural Networks (DNNs) [10]. However a quick look at the standard recipes of Kaldi show that, its training tasks are mostly built to be executed in CPU and GPU clusters whereas CMU Sphinx can train reasonably detailed acoustic models in desktop systems. In addition, although DNN based models are the state-of-the-art, PocketSphinx holds the advantage of being easily deployable in resource limited environments such as hand-held devices. Unfortunately due to its dependence in external libraries such as LAPACK, Kaldi's usage in hand-held devices end-up being problematic due to memory limitations [11].

3. Data collection and preparation

We have decided to exploit readily available data on the Internet for compiling the necessary training data. We found this approach to be less costly compared to approaches that involve script preparation and recording.

3.1. Raw data collection

In order to gather Catalan audio with transcriptions, we have taken advantage of the fact that Catalan public television makes its programs accessible online for the general public[2]. Firstly we have revised the full list of available programs and eliminated the ones which have inappropriate subtitles; such as the live interviews, talk shows and daily news which necessitate the captioning to be done spontaneously. Effectively these captions are not synchronized and approximate hence those programs are not considered. Secondly we have chosen the programs with modalities of speech that are useful for ASR training. This meant looking for programs with natural conversational language such as interviews or programs with clear dictation with minimum amount of background noise or music. Based on these priorities, we downloaded approximately 490 hours of video with their corresponding subtitles in *srt* format from 17 programs, the distribution of topics and durations for each program can be seen in the table 1.

[2] http://www.ccma.cat/tv3/

3.2. Acoustic data preparation

For training the acoustic models for an ASR system, audio segments with lengths between 5-20 seconds are needed. The video files we have downloaded have lengths ranging from approximately 7 minutes to 75 minutes. In order to cut these video files to meaningful segments with desired lengths, we have taken advantage of the subtitling.

Before undertaking the segmentation process, as a start, we first had to verify that the subtitle files correspond correctly to the video recordings. In this stage we did not assess the quality of the subtitle cues, but rather checked for more general problems, which we have determined to be three; (1) if the downloaded subtitle file is empty, (2) if the subtitles correspond to a different program, or (3) if the subtitles correspond to the correct video but with wrong timestamps, possibly due to a wrong frame per second assumption made during the export. In order to solve these problems, we first checked whether the subtitle file had cues or not. If it had, we looked at the end time stamp of the last cue and compared it with the duration of the video. In the case of an empty subtitle file, or subtitle cues extending beyond the video duration, we have eliminated the video and subtitle from further processing.

The training segments are defined as word sequences which start and end with silences. We have used the minimum separation duration of 100 ms, and any sequential cues with separations smaller than this amount are grouped together in batches. Our assumption was that longer separations are more likely to signal the beginning and the end of sentences or simply are most like to represent the pauses in speech, hence practically defining the start and the end of the audio segments. After the cues are grouped into batches, their durations are further evaluated and in the case the durations did not fall between 5 and 20 seconds they were eliminated.

Before starting the actual segmentation of the video files, the text content is further filtered. This included getting rid of the explanatory captions; i.e. any content within parenthesis or brackets, and cues starting with #. Additionally, we have eliminated the cues which had content presumably not in Catalan. As a final step we have normalized the text by eliminating the punctuation and symbols. There are exceptions in this stage due to the nature of Catalan morphology, since dashes and apostrophes might be necessary combining pronouns with verbs.

Finally, the complete audio that is extracted from the video file is cut according to this list of the calculated and filtered segments. During the actual segmentation process, we convert the audio files into mono channel with a sampling of 16 kHz.

3.3. Text and language data preparation

Before starting the actual training process using the acoustic data, a phonetic lexicon needs to be built matching each word as they appear in the transcriptions with its phonetic description. This was achieved by using the grapheme to phoneme conversion tool embedded in the rule based speech synthesis tool *espeak*. To construct a phonetic lexicon, we first extracted all the unique words in the transcriptions, converted them to their IPA (International Phonetic Alphabet) written format and

Table 1: *The programmes used in constructing the ASR system. Table shows their respective themes, total downloaded durations and the final duration used for the training.*

Programme name	**Theme**	**Downloaded Duration**	**Filtered Duration**
Zona UFEC	Sports	22:42:22	8:31:20
Art Endins	Outreach	6:22:9	2:04:31
El dia de demà	Current events	8:55:59	3:22:25
Quan arribin els marcians	Culture (Current Events)	17:45:38	7:26:05
Tot un món	Current events	5:20:12	1:39:16
Valor afegit	Current events	47:45:34	26:20:37
Afers Exteriors	Current events, Outreach	53:01:55	30:31:17
A la presò	Outreach	5:55:23	0:53:01
Arts i Oficis	Outreach	12:42:31	6:00:14
Benvinguts a l'hort	Gastronomy, Outreach	11:35:16	6:58:15
Collita pròpia	Outreach	10:01:30	6:09:03
Catalunya des del mar	Current events, Outreach	5:40:51	2:07:23
De llit en llit	Outreach	3:00:33	1:02:05
Detectiu	Outreach, Fiction	6:47:56	2:22:11
Economia en colors	Entertainment	16:03:59	7:44:17
Els dies clau	Outreach	6:19:19	3:23:36
Fotografies	Outreach	5:31:49	2:29:39
Generació Selfie	Outreach	6:51:19	2:51:46
Històries de Catalunya	Outreach	11:28:34	5:37:31
La salut al cistell	Gastronomy, Outreach	12:04:46	7:49:46
L'ofici de viure	Outreach	4:19:38	1:26:45
Millennium	Outreach	29:25:58	16:11:53
Trinxeres	Outreach	7:15:55	2:37:20
Programa sindical CCOO	Outreach	12:16:42	7:03:45
Quèquicom	Outreach	52:23:31	27:10:41
Sota terra	Outreach, Entertainment	11:36:03	4:07:41
Veterinaris	Outreach	49:42:21	21:20:23
Via llibre	Outreach, Entertainment	46:59:25	24:52:51
Total		489:57:23	240:15:52

finally converted these IPA version to the CMU Sphinx readable format. In total we have used 37 phonemes, consistent with the literature on Catalan phonetic corpus [12].

Additional information on the structure of Catalan language is necessary for the decoding phase. Within an ASR system the statistical information on the linguistic grammar and syntax represented through the language model, and these models can be prepared using a sufficiently large text corpus. In this work, we have taken advantage of the subtitle of our audio corpus and merged them with the Catalan OpenSubtitles Corpus [13] to build a basis for our language models.

The final corpus is cleaned from all symbols and punctuation, and numbers are normalized using *espeak* tool. The complete corpus has 5.3 million tokens with around 100,000 unique tokens which we used for compiling the phonetic lexicon. Using approximately 58k words (which appear at least twice in the corpus) we have prepared one 3-gram (OT_large_3gram) and another 4-gram (OT_large_4gram) language model in ARPA format using the CMU Language Model toolkit (CMUCLMTK).

4. Training

For our training process we have used the standard CMU Sphinx training steps with very minor changes. The training starts with extraction of the Mel-Frequency Cepstral Coefficients (MFCC) between 130 and 6800 Hz; i.e. 12 cepstra using the C0 as the energy component plus their deltas and delta deltas adding up to 39 total parameters (1s_c_d_dd). For the acoustic model training, our Gaussian mixture model contains 32 Gaussian densities, and 6000 tied HMM states.

In our process, we started by estimating the transition probabilities of the Context-Independent (CI) HMMs for forced alignment of the acoustic data. For the forced alignment itself we used the *sphinx3_align* executable that needed to be compiled apart from the Sphinx-5prealpha library. In this step, the audio files are aligned with their respective transcriptions using the CI models, in the case when there is a mismatch with the transcription and the alignment result, the audio files are eliminated for the following steps. After the non-aligning segments are eliminated we were left with 240 hours of total audio. The final amount of audio per programme used is shown in table 1.

The transition probabilities of the CI HMMs re-estimated using filtered data set, and following this phase a complete list of tri-phones (58289 in our case) are built and their transition probabilities are estimated in the form of Context-Dependent (CDs) HMMs. These tri-phones account for both between-word and within-word contexts, however since the training data might not account for all the possibilities, the unseen tri-phones are tied to the seen tri-phones using decision trees.

We performed our training in a resource limited environment. For four threads of Intel(R) Atom(TM) CPU N2800 with

1.86GHz, the whole training process took about 120 hours.

4.1. Evaluation

In order to evaluate the word error rate (WER) of the acoustic models we wanted to make sure that the test voices do not appear in the training corpus. In order to evaluate the acoustic models for similar recordings, we downloaded different *TV3* programmes that were not used in the training. 4 hours of new *TV3* recordings evaluated with the OT_large_4gram language models resulted in a WER of %35,2. For the decoding we used the standard decoding script within the CMU Sphinx.

In order to evaluate the accuracy of the models in a cleaner environment and also to guarantee 100% speaker exclusion we decided to use another test set for evaluation round. For this we used FESTCAT corpus, which is specifically designed for creating a speech synthesis system for Catalan [14, 15], and consists of 28 hours of recordings from 10 different voices (5 female, 5 male). For evaluating our ASR system we used 4 total hours of 4 female and 4 male voices with their corresponding transcripts. It should be noted that due to the clean environment of the recordings, the FESTCAT dataset also represents a more ideal audio quality.

Due to our restricted text corpus, we have created another set of language models in addition to the ones explained in the subsection 3.3 specifically for the test decoding. The second set of language models uses a corpus of FESTCAT text plus the corpus explained in the subsection 3.3. Using this new corpus we created one 3-gram language model with the most frequent 20k words (OTF_20k_3gram) and two other models with the most frequent 58k words (OTF_large_3gram, OTF_large_4gram) similar to the OT_large models. Whereas for the OT_large_4gram language model we ended up with a WER of 31.95%, the best results were attained by the OTF_large_4gram model at 11,68%. The results for each language model with the corresponding real-time decoding factor (xRT) for an Intel i7-4510U 3 Ghz Quad-core architecture is shown in the table 2. The high precision OTF results, show that if the acoustic conditions are perfect and the language models are "in-domain," our acoustic models can recognize voices that are not in its training set reasonably well. In addition, for our cases the main factor which improved the recognition precision was the amount of pruning that the corpus was subjected to for constructing the language models. Whereas moving from the most frequent 20k words to most frequent 58k words makes a considerable improvement, the effect of using 4-gram instead of 3-gram seems to be very small, probably due to our specific test condition. However one important difference between the 3-gram and 4-gram models is the xRT, for which the 3-gram models are considerably faster than the 4-gram models. Note that we did not undertake any optimization of the decoding parameters neither for best precision not for the best computational performance.

Table 2: *The WER and xRT results for different language models for the FESTCAT test dataset.*

Language Model	**WER** (%)	**xRT**
OT_large_4gram	31,95	0.952
OTF_20k_3gram	22,50	0.872
OTF_large_3gram	12,11	0.900
OTF_large_4gram	11,68	1.002

5. Future Work

The most important and basic step for improving our ASR system is to use a better pronunciation dictionary using a better grapheme to phoneme conversion system. In this work, for its ease of use we have taken advantage of *espeak*, however the festival based FESTCAT speech synthesis system is specifically implemented for Catalan and allows for a more refined grapheme to phoneme conversion. Training the acoustic model with this improved pronunciation dictionary will allow for better results overall.

For the acoustic data itself depending on the sound quality, background music levels and the speaker mistakes, it should be clustered into clean and other, similar to the *librispeech* dataset [16]. Additionally, we plan to do a gender diarization model, determining whether the voice is male of female for each segment, in order to assess the gender balance of the whole dataset.

With these acoustic models, it will be possible to do alignment of an audio with its given text. This process will not only be useful in cleaning the dataset itself, but also will allow extending the current set without relying on the cue start and end times within the subtitles. This implies further Catalan acoustic data could be assembled by using the audio and just its corresponding transcription.

Related to this possibility, another tool we would like to develop is a system of automatic punctuation in Catalan. The readability of recognized transcripts depend a lot on sentence segmentation and correct punctuation. The methods for training punctuation engines using recurrent neural networks (RNN) are very well developed, especially with the use of a large text corpus [17]. But also recently it was demonstrated that acoustic data with word-aligned transcriptions can be used to create prosody based punctuation models [18]. For now this type of models have only been trained for English. With the possibility of doing word level alignments for Catalan, we will be able to train one in Catalan in the recent future.

6. Conclusions

In this paper, we have described building of an ASR system for a new language, using only publicly available resources. Applying our methodology on Catalan, we compiled a dataset of 240 hours of transcribed broadcast speech and used it to develop large-vocabulary speech recognition models, both of which are distributed openly online. The accuracy of the resulting models show that they can be a base for speech technology developers to access the Catalan speaking community. Building a voice input interface for a desktop or mobile application is easy as installing the CMU Sphinx toolkit[3] and placing the models in its installation directory. The ASR system further gives the possibility to adapt acoustic and language models for more specialized vocabularies and acoustic environments. We believe that the practical and low-cost setup of CMU Sphinx makes it an important player amongst other ASR engines, despite the more modern neural network based alternatives. It keeps its relevance especially for minority languages which have little open acoustic data resources available.

7. Acknowledgements

This project was funded by Softcatalà. The authors would like to thank Antonio Bonafonte for his guidance during the writing of this paper.

[3] https://cmusphinx.github.io/wiki/tutorial/

8. References

[1] H. Schulz, M. Ruiz, and J. A. R. Fonollosa, "TECNOPARLA - Speech technologies for Catalan and its application to speech-to-speech translation," *Procesamiento del lenguaje natural*, vol. 41, pp. 319–320, Sep 2008.

[2] J. Mariño, J. Padrell, A. Moreno, and C. Nadeu, "Workshop on speech recognition based on very large telephone speech databases," C. Draxler, 2000, pp. 57–61.

[3] M. Gales, S. Young *et al.*, "The application of hidden markov models in speech recognition," *Foundations and Trends in Signal Processing*, vol. 1, no. 3, pp. 195–304, 2008.

[4] K.-F. Lee, H.-W. Hon, and R. Reddy, "An overview of the sphinx speech recognition system," in *Readings in speech Recognition*. Elsevier, 1990, pp. 600–610.

[5] P. Placeway, S. Chen, M. Eskenazi, U. Jain, V. Parikh, B. Raj, M. Ravishankar, R. Rosenfeld, K. Seymore, M. Siegler *et al.*, "The 1996 hub-4 sphinx-3 system," in *Proc. DARPA Speech recognition workshop*, vol. 97. Citeseer, 1997.

[6] P. Lamere, P. Kwok, E. Gouvea, B. Raj, R. Singh, W. Walker, M. Warmuth, and P. Wolf, "The CMU sphinx-4 speech recognition system," in *IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP 2003), Hong Kong*, vol. 1, 2003, pp. 2–5.

[7] D. Huggins-Daines, M. Kumar, A. Chan, A. W. Black, M. Ravishankar, and A. I. Rudnicky, "Pocketsphinx: A free, real-time continuous speech recognition system for hand-held devices," in *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, vol. 1. IEEE, 2006, pp. I–I.

[8] D. Huggins-Daines and A. I. Rudnicky, "Mixture pruning and roughening for scalable acoustic models," in *Proceedings of the ACL-08: HLT Workshop on Mobile Language Processing*, 2008, pp. 21–24.

[9] ——, "Combining mixture weight pruning and quantization for small-footprint speech recognition," in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*. IEEE, 2009, pp. 4189–4192.

[10] C. Gaida, P. Lange, R. Petrick, P. Proba, A. Malatawy, and D. Suendermann-Oeft, "Comparing open-source speech recognition toolkits," *Tech. Rep., DHBW Stuttgart*, 2014.

[11] P. Vojtas, J. Stepan, D. Sec, R. Cimler, and O. Krejcar, "Voice recognition software on embedded devices," in *Asian Conference on Intelligent Information and Database Systems*. Springer, 2018, pp. 642–650.

[12] I. Esquerra, C. N. Camprub, L. Villarrubia, and P. Len, "Design of a phonetic corpus for speech recognition in Catalan," in *Workshop on Language Resources for European Minority Languages at the Conference on Language Resources and Evaluation (LREC)*, Granada, 1998.

[13] J. Tiedemann, "News from OPUS - A collection of multilingual parallel corpora with tools and interfaces," in *Recent Advances in Natural Language Processing*, N. Nicolov, K. Bontcheva, G. Angelova, and R. Mitkov, Eds. Borovets, Bulgaria: John Benjamins, Amsterdam/Philadelphia, 2009, vol. V, pp. 237–248.

[14] A. Bonafonte, J. Adell, I. Esquerra, S. Gallego, A. Moreno, and J. Pérez, "Corpus and voices for Catalan speech synthesis," in *Proceedings of LREC Conference 2008*, 2008, pp. 3325–3329.

[15] A. Bonafonte, L. Aguilar, I. Esquerra, S. Oller, and A. Moreno, "Recent work on the FESTCAT database for speech synthesis," in *Proceedings of LREC Conference 2008*, 2009, pp. 3325–3329.

[16] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an ASR corpus based on public domain audio books," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 5206–5210.

[17] O. Tilk and T. Alumäe, "Bidirectional recurrent neural network with attention mechanism for punctuation restoration," in *Interspeech 2016*, 2016.

[18] A. Öktem, M. Farrus, and L. Wanner, "Attentional parallel RNNs for generating punctuation in transcribed speech," in *5th International Conference on Statistical Language and Speech Processing SLSP 2017*, Le Mans, France, 2017.