JULIUS ASR FOR ITALIAN CHILDREN SPEECH

Giulio Paci, Giacomo Sommavilla, Fabio Tesser, Piero Cosi
ISTC CNR - UOS Padova
Istituto di Scienze e Tecnologie della Cognizione
Consiglio Nazionale delle Ricerche - Unità Organizzativa di Supporto di Padova, Italy
[giulio.paci, giacomo.sommavilla, fabio.tesser, piero.cosi]@pd.istc.cnr.it

1. ABSTRACT

This work describes the use of the Julius ASR engine for Italian children speech recognition in child-robot interactions within the European project called ALIZ-E, "Adaptive Strategies for Sustainable Long-term Social Interaction". The goal of the project is to develop embodied cognitive robots for believable any-depth affective interaction with young users over an extended and possibly discontinuous period. Speech recognition plays an important role whenever the verbal interaction between the robot and the child user is crucial for the ALIZ-E experiments. This work focuses on the Quiz Game ALIZ-E scenario, where the child and the robot ask each other general knowledge questions.

Julius' low system requirements and small memory footprint makes it an excellent candidate for implementing speech recognition into a real-time integrated system handling several components, like the ALIZ-E one. Also, with Julius, it has proven to be very easy to integrate the desired features into the system, because of its simple API.

The Italian FBK ChildIt corpus has been used to train the acoustic model for the system, while a very simple target-specific language model has been created using the questions and answers database of the Quiz Game ALIZ-E scenario. Preliminary results on real data recorded in Wizard of Oz setup reports an average of 75.7% correct words recognition rate, 10.4% inserted words and 34.7% WER.

2. INTRODUCTION

The ALIZ-E project [1] aims to extend the science and technology behind long-term human-robot interaction, with a specific goal of supporting children engaged in a residential and hospital diabetes management course.

Since speech is the most natural and used mode of communication in many circumstances, part of the project is dedicated to developing conversational systems capable to understand the children's spoken input. In this context, it has become necessary to develop an Automatic Speech Recognition (ASR) system usable by the Italian children involved on the experimental part of the project.

Developing components for robust spoken input processing for children is thus of crucial importance in ALIZ-E. However, this is not a trivial task: there are not so many speech corpora available to train speech recognition models for children and the acoustic and linguistic characteristics of child speech differ widely from those of adult speech, not only at the frequency range level but also in the production modalities (Gerosa et al., 2009).

For example, there are more disfluencies not only in spontaneous speech but even in read speech. An investigation concerning these problems (Potamianos & Narayanan, 2003) has shown that word error rate for children is normally higher than that for adults even when using an acoustic model trained on child speech.

[1] http://www.aliz-e.org/.

Moreover, children's vocal tract lengths can differ a lot from one child to another. For this reason adaptation technologies like VTLN (Zhan & Waibel, 1997) can be very useful in the ALIZ-E project. Since each child will interact several times with the robot we can consider using data from previous interactions to adapt the models.

The ALIZ-E integrated system (Kruijff-Korbayova et al., 2012) implemented different game-like activities that a child can undertake. Among those activities, the most challenging one for what it concerns ASR is the Quiz Game.

The paper is organised as follows: Section 3 introduces the Quiz Game, Section 4 describes the open-source Julius ASR engine integrated into the ALIZ-E platform. The development of children Acoustical Model and Language Model for Italian are described in Section 5 and Section 7. An experiment about the use of VTLN is reported in Section 6. Results about the use of these resources on a test set of the ALIZ-E Quiz Game corpus are described in Section 8. Finally, Section 9 concludes the paper.

3. QUIZ GAME DESCRIPTION

The quiz game is one of the several games implemented in an integrated system that combines several components to allow a humanoid robot to interact with children in the context of the EU-FP7 Project ALIZ-E. In this game the verbal interaction is crucial as no other input interface is used.

The game starts with the robot explaining the rules: either the robot or the child should provide multiple choices questions to the other player. After each answer, the asker is supposed to provide feedback about the correctness of the answer and, if necessary, give a second possibility to the answerer. After three questions, the robot summarizes the results and the players switch their roles. When each player has played both roles, a round is completed and the players have to agree if they want to continue playing or not. The game ends when one of the two players wants to leave.

Figure 1 shows the setup of the interaction. The child wears a close talk microphone and sits in front of the robot at a distance of approximately 40 cm. The microphone is a hand-free radio close talk microphone (Proel© RM300H, Radio frequency range: UHF High Band 750-865 MHz, Microphone: headset HCM-2), selected to reduce its invasiveness during the child-robot interaction. The microphone has been connected to a Zoom H4n audio recorder, used as an USB audio input interface. A tablet, connected via wireless to the integrated system, is used to display Quiz questions and answers to the asking user. The tablet is mounted on a support which allows to rotate the display, so that it is possible to show information to both players.

The Questions & Answers rounds have the following features:

- the robot asks the child questions from the domains of diabetes and nutrition mixed with questions taken from a Trivial Pursuit game for children; the child chooses questions from a set of cards, just like in a standard Trivial Pursuit game;

- the questions are presented as multiple choice questions, and the child answers by providing the answer or the corresponding number (or letter);

- the system provides correctness feedback after each child answer and expects feedback from the child when it is providing answers: a correct answer gets positive feedback, an incorrect answer gets feedback that elicits another response; if the second answer is wrong, the system provides the correct answer; feedback consists of a combination of verbal and non-verbal signs.

At all times the child can ask the robot to stop (this will give our measure of interaction time). The child can also ask the robot to repeat a question or possible answers. Also, the system is designed to display familiarity in several ways (addressing the child by their name, referencing to previous achievements in the game, e.g., the number of correctly answered quiz questions).

3.1. Experiments for data collection

So far, we have gathered speech data from non-autonomous interactions with the children. These experiments are characterized by the fact that recognition and understanding of user input are not automatic, but performed by a human operator. The personnel of the non-autonomous experiments consists of:

- an experimenter, who welcomes the child, introduces him/her to the interaction and takes care of submitting questionnaires at the end of the game;
- a "Wizard of Oz" experimenter, who remotely controls the system, by entering user input data.

Even if most quiz questions and answers' options are read by the children, speech data from these interactions can be classified as spontaneous, since there are a lot of mispronunciations, false starts and fillers.

Within the ALIZ-E project consortium, it has been agreed to manually annotate speech recordings from the above mentioned interactions. In particular, a special category called "domain objects" has been defined. Domain objects are speech events relevant for the Natural Language Understanding (NLU) component (ALIZ-E Team, 2013), i.e. sentences that carry special meaning, important to specific ALIZ-E scenarios. In the case of the Quiz Game, the two main labels that have been used to mark the domain objects are *Question*, used when the child poses a Quiz question (e.g. "Chi allattò Romolo e Remo?") and *Answer*, used to mark a Quiz answer option (e.g. "Una lupa", "La loro mamma", "La prima"). Each domain object can be tagged as "unappropriate" (for example when the provided answer was not in the answers' list) and/or "incomplete" (if the user did not utter the sentence completely).

These interactions' audio files have been transcribed, using the Transcriber [2] software, at different levels: speech, interruptions, domain objects, noise and fillers. The Trsdump tool has been developed in order to parse the annotation files and to automatically create different data sets with peculiar features (e.g., suitable for ASR/NLU testing).

So far, 8 interactions transcriptions have been completed (speech, fillers and domain objects have been annotated) and revised, while 68 have been completed, but still need proper revision. Globally, the 76 transcriptions consist of about 20 hours of total audio, containing more or less 3.6 hours of actual children speech.

These data have been used to test speech recognition adaptation techniques, as reported in Section 6. We intend to use those data to perform batch (offline) ASR experiments and improve recognition, by incorporating non verbal sounds found in speech data from interactions into AM and LM. When results are satisfactory enough to declare ASR reliable, it will be included in an autonomous system.

[2] Transcriber: `http://trans.sourceforge.net/en/presentation.php`

Figure 1: Screenshots taken from the video recordings of the experiment interactions.

4. JULIUS DESCRIPTION

Open-Source Large Vocabulary Continuous Speech Recognition Engine Julius (A. Lee et al., 2001) is a high-performance ASR decoder for researchers and developers, designed for real-time decoding and modularity.

Julius is particularly suitable for the ALIZ-E project. In fact, it has proven to be very easy to implement and incorporate the desired features into the integrated system, because its decoder API is very well designed and the core engine is a separate C library. Moreover, Julius has low system requirements, a small memory footprint, a high-speed decoding and it can swap language models at run-time: all these features are crucial in a real-time integrated system handling several components. Its configuration is modular (i.e., each configuration file can embed another one covering only one particular aspect of the configuration).

Most of the features available in other state-of-the-art decoders are also available for Julius, including major search techniques such as tree lexicon, N-gram factoring, cross-word context dependency handling, enveloped beam search, Gaussian pruning, Gaussian selection, etc. Finally, Julius integrates an GMM- and Energy-based VAD.

We tried also CMU Sphinx-3 (K.-F. Lee et al., 1990) for speech recognition. However it has been difficult to implement live decoding and run-time features with it and Sphinx-3 upstream code is no longer maintained. So Sphinx-3 has been replaced with Julius as the ASR engine in our recognition experiments with children speech.

4.1. Recognition output

The ALIZ-E Julius component can produce Speech recognition output as an n-best list (i.e., the set of the *n* most probable sentences) or a lattice. A lattice (or Word Graph in Julius terminology) is an acyclic ordered graph, in which nodes represent words and edges represent transition probabilities (weighted by acoustic and language model scores). ASR hypothesis can be expressed as a Word Graph, that is a more powerful tool than n-best list for Spoken Language Understanding (SLU), since lattices provide a wider set of hypothesis from which to choose and a more accurate representation of the hypothesis space.

4.2. Multi-model recognition

Julius supports multi-model recognition, as explained in (Lee, Akinobu, 2010). This means that $n > 1$ configuration instances can be loaded and the Julius engine will output n results for a single audio input at the same time.

To enable multi-model recognition, multiple search instances must be declared. A search instance is defined within a Julius configuration file, and links to an AM and an LM, with custom recognition parameters. Every ASR result comes from a single search instance.

The multi-model recognition can be used in the ALIZ-E project in order to improve accuracy of the NLU module. Using multiple search instances at a time, one can keep the result from the search instance that has the greatest likelihood, or the one that is related to a specific modality. For example, the result for an instance associated with an LM built on a “greetings” set of sentences can be given higher confidence, because another component in the integrated system (such as the Dialogue Manager) expects that kind of communication from the user in that particular moment.

Moreover, thanks to this feature, in the future it will also be possible to create models for input rejection, so that unwanted speech events can be detected and discharged as needed.

5. ITALIAN ACOUSTIC MODEL

In order to create the acoustic model, the Italian FBK ChildIt Corpus (Gerosa et al., 2007) has been used. The corpus consists of Italian children voices, counting almost 10 hours of speech from 171 children; each child reads about 60 children literature sentences; the audio was sampled at 16 kHz, 16 bit linear, using a Shure SM10A head-worn mic.

The LVCSR Engine Julius distribution does not include specific training tools for acoustic models, however any tool that create acoustic models in the Hidden Markov Model Toolkit

(HTK) format can be used. The HTK tools (Young et al., 2006) have been used for this task in the ALIZ-E project, following the Voxforge HTK training for Julius tutorial (VoxForge, 2012). Moreover, a procedure to build multi-gaussian Acoustic Model (AM) has been implemented with HTK. The procedure to compute multi-gaussian AM takes as input a single-gaussian model and consists of the following steps: perform forced alignment with single-gaussian model; estimate parameter with an n-gaussian ($n > 1$) model; perform forced alignment with the model computed in the previous step; estimate parameter with an m-gaussian ($m > n$) model.

Firstly, the multi-gaussian parameter estimation has been tested with 9 gaussian per state. Finally, a good tradeoff between accuracy results and computational performance has been reached with a multi-gaussian model estimated with 32 gaussians for the first state, 16 gaussians for the second state and 32 gaussians for the third state. The central state has fewer gaussians, assuming that in the central state speech features are more stable.

6. ADAPTATION EXPERIMENTS

Julius supports on-the-fly Input/AM normalisation: cepstral mean normalisation (CMN), cepstral variance normalisation (CVN), vocal tract length normalisation (VTLN).

VTLN (Young et al., 2006; Zhan & Waibel, 1997) has been used to improve acoustic model results: five recognitions, with five different configurations for VTLN parameters run simultaneously. The configuration providing the best confidence score is selected to provide results. In order to decide the number of configurations and parameters, a grid search experiment has been carried on children speech training data: a gender independent AM has been trained using children voices as reported in Section 5.

Forced alignment has been run on all training sentences (more than 10000) with alpha values spanning from 0.7 to 1.3 (with a 0.1 step). Other VTLN warp function parameters are 50 and 3000 Hz for low and high cutoffs. For each training file we selected the alpha value that maximizes the acoustic score. We expected most optimal alpha values to fall around the value of 1, and we verified this hypothesis, as shown in Table 1.

alpha	0.7	0.8	0.9	1.0	1.1	1.2	1.3
files	1	61	2216	6158	1741	279	6

Table 1: alpha values selected with respect to training files.

This means that alpha values below 0.7 and above 1.3 would be very unlikely, and even these two values are not very likely. We noticed that the acoustic score is a convex function as alpha changes, i.e., for the optimal alpha value of every sentence, the first derivate of the logarithmic acoustic score (as a function of alpha) is a monotonic crescent function. Hence the optimal alpha value is a maximum of a concave acoustic score function, since probability, between 0 and 1, leads to $log(P) < 0$. Finally, we observed very little fluctuation of alpha values within each speaker. In a later experiment we tried a different grid search granularity, starting from alpha = 0.8, step range = 0.05 and ending at alpha = 1.2. With respect to the previous test, many more speakers fluctuate between more than one alpha value (some speakers span to 3 values). The acoustic score does not change much between the best and the second best (on average, there is less than 1% difference between the two). Still, the acoustic score is a convex function of alpha, so either the best or the second best score belongs to the 0.1 grid. Thus we decided that a step range of 0.05 was not worth the complexity increase.

7. ITALIAN LANGUAGE MODEL

The LVCSR Engine Julius supports N-gram, grammar and isolated word Language Models (LMs). Also user-defined functions can be implemented for recognition.

However its distribution does not include any tool to create language models, with the exception of some scripts to convert a grammar written in a simple language into the Deterministic Finite Automaton (DFA) format needed by the engine. This means that external tools should be used to create a language model.

7.1. N-gram LM

The Julius engine supports N-gram LMs in ARPA format. SRI-LM toolkit (Stolcke, 2002) has been used to train a 4-gram model for question recognition of the Quiz Game ALIZ-E scenario. The Quiz questions and answers database has been used as training material for a "question recognition" model. The model is very simple and limited, but it should be enough to properly recognise read questions, because the questions to be recognised are expected to be in the training set.

7.2. Grammar LM

The Julius engine distribution includes some tools that allow to express a Grammar in a simple format and then to convert to the DFA format needed by Julius. That format, however, has very few constructs that help writing a proper grammar by hand and writing a non-trivial grammar is very hard. Third-party tools exist to convert an HTK standard lattice format (SLF) to the DFA format and to optimise the resulting DFA (Julius development team, 2012). SLF is not suitable to write a grammar by hand, but HTK provides tools that allow a more convenient representation based on the extended Backus-Naur Form (EBNF) (Young et al., 2006). A simple model for Quiz answers recognition has been written in the EBNF-based HTK grammar language. Part of the grammar was automatically derived by including the answers in the Quiz database. Rules were added to handle common answers and filler words.

8. RESULTS

ID	**# Snt**	**# Wrd**	**WCR [%]**	**Sub%**	**Del [%]**	**Ins [%]**	**WER [%]**
1	4	22	86.4	13.6	0.0	13.6	27.3
2	6	82	76.8	22.0	1.2	23.2	46.3
3	5	40	70.0	25.0	5.0	17.5	47.5
4	7	62	75.8	12.9	11.3	4.8	29.0
5	15	114	93.9	4.4	1.8	5.3	11.4
6	4	49	65.3	30.6	4.1	12.2	46.9
7	12	106	58.5	35.8	5.7	4.7	46.2
8	11	84	70.2	23.8	6.0	9.5	39.3
Total	64	559	74.6	20.9	4.5	10.2	35.6

Table 2: Preliminary ASR results on quiz question recognition.

Table 2 shows the results of ASR applied to 64 utterances (559 words), where a child poses quiz questions to the NAO robot. In this setup the language model described in Section 7.1 and the acoustic model described in Section 5 have been used, without any form of speaker adaptation. Table 3 shows the results of the same setup, but using VTLN techniques. On average the setup with VTLN gives better results, improving all the metrics with

ID	# Snt	# Wrd	WCR [%]	Sub%	Del [%]	Ins [%]	WER [%]
1	4	22	86.4	13.6	0.0	22.7	36.4
2	6	82	80.5	18.3	1.2	24.4	43.9
3	5	40	72.5	27.5	0.0	22.5	50.0
4	7	62	79.0	14.5	6.5	1.6	22.6
5	15	114	93.9	4.4	1.8	4.4	10.5
6	4	49	65.3	26.5	8.2	12.2	46.9
7	12	106	59.4	34.0	6.6	4.7	45.3
8	11	84	69.0	25.0	6.0	8.3	39.3
Total	64	559	75.7	20.2	4.1	10.4	34.7

Table 3: Preliminary ASR results on quiz question recognition (VTLN).

the exception of the insertion rate (Ins). Comparing results file by file, the Word Correct Rate (WCR) is always equal or better in the VTLN setup, while the Word Error Rate (WER) is worse for files 1 and 3. By manual inspection we tried to classify and identify the reason of the errors. 21 errors were due to Out-Of-Vocabulary words, mostly caused by repetitions, misreadings and interrupted words. There were 13 real insertions (i.e., words that were not related to any sound): these are monosyllabic words, inserted for unknown reasons. There was only one word “essere” inserted in one transcription, due to the robot speaking between two child’s words. In five cases the children misread some monosyllabic words and they were recognized as those in the original sentence. In all the other cases the words have been recognized as sequences of words which sounds very similar. In 9 cases these errors were originated by a mismatch between the actual pronounce and the phonetic dictionary.

With the standard setup, on average, we get 74.6% correct words, 10.2% inserted words and 35.6% WER, while with the VTLN setup we get 75.7% correct words, 10.4% inserted words and 34.7% WER. Taking the ASR hypothesis as input to a specific Natural Language Understanding (NLU) module, specifically designed and implemented for ALIZ-E, questions were correctly identified by fuzzy matching against the quiz database contents.

This is an encouraging first result and further experiments will show whether this level of ASR+NLU performance suffices to sustain the interaction.

9. CONCLUSIONS

This work described the efforts spent on the task of Italian children’s speech recognition within the European project ALIZ-E. Since this project aims to study children-robot interactions, an integrated system has been built with several software components developed by the many partners of the project.

One of the main activities designed for interaction is a Trivial Pursuit-like quiz game, where the user and the robot take turns asking questions each other. Verbal interaction is a fundamental aspect in the Quiz Game, so speech recognition plays a crucial role.

This paper described the integration of the Julius ASR component in the ALIZ-E system and the work dedicated to improve and evaluate it, by: 1) building an acoustic model for Italian children speech; 2) building a linguistic model for the ALIZ-E Quiz game scenario; 3) gathering data of real case interactions and annotating them; 4) carrying on speech recognition adaptation experiments. Future works include testing the Julius component in an autonomous system.

ACKNOWLEDGEMENTS

Parts of the research reported on in this paper were performed in the context of the EU-FP7 project ALIZ-E (ICT-248116).

REFERENCES

ALIZ-E Team. (2013). Deliverable D4.3 adaptive HRI with comprehension and production for discussing encounters (Tech. Rep.). ALIZ-E Adaptive Strategies for Sustainable Long-Term Social Interaction.

Gerosa, M., Giuliani, D., & Brugnara, F. (2007, Feb), Acoustic variability and automatic recognition of children's speech, Speech Communication, Vol. 49, 847–860.

Gerosa, M., Giuliani, D., Narayanan, S. S., & Potamianos, A. (2009, Feb), A review of ASR technologies for childrens speech, in Proceedings of Workshop on Child, Computer and Interaction (WOCCI), Vol. 49, 847–860.

Julius development team. (2012, March). Open-source large vocabulary CSR engine julius. Retrieved from `http://julius.sourceforge.jp/`

Kruijff-Korbayova, I., Cuayahuitl, H., Kiefer, B., Schröder, M., Cosi, P., Paci, G., … Verhelst, W. (2012), Spoken Language Processing in a Conversational System for Child-Robot Interaction, in Proceedings of Workshop on Child, Computer and Interaction (WOCCI).

Lee, A., Kawahara, T., & Shikano, K. (2001), Julius - an open source real-time large vocabulary recognition engine, in Proceedings of European Conference on Speech Communication and Technology (EUROSPEECH), 1691-1694.

Lee, K.-F., Hon, H.-W., & Reddy, R. (1990, January), An overview of the SPHINX speech recognition system, IEEE Transactions on Acoustics, Speech and Signal Processing, Vol. 38, no. 1, 35 - 45.

Lee, Akinobu. (2010, May). Juliusbook. Retrieved from `http://sourceforge.jp/frs/redir.php?m=jaist&f=%2Fjulius%2F47534%2FJuliusbook-4.1.5.pdf`

Potamianos, A., & Narayanan, S. (2003, November), Robust recognition of children's speech, IEEE Transactions on Speech and Audio Processing, Vol. 11, no. 6, 603–616. Retrieved from `http://dx.doi.org/10.1109/tsa.2003.818026`

Stolcke, A. (2002), SRILM - an extensible language modeling toolkit, in Proceedings of eventh International Conference on Spoken Language Processing (ICSLP), ISCA, 901–904.

VoxForge. (2012, March). Tutorial: Create acoustic model - manually. Author. Retrieved from `http://www.voxforge.org/home/dev/acousticmodels/linux/create/htkjulius/tutorial`

Young, S. J., Evermann, G., Gales, M. J. F., Hain, T., Kershaw, D., Moore, G., … Woodland, P. C. (2006), The HTK book, version 3.4, Cambridge, UK: Cambridge University Engineering Department.

Zhan, P., & Waibel, A. (1997). Vocal tract length normalization for large vocabulary continuous speech recognition (Tech. Rep.). CMU COMPUTER SCIENCE TECHNICAL REPORTS.