The CSU-K Rule-Based Pipeline System for Spoken CALL Shared Task

Nico Axtmann, Carolina Mehret, Kay Berkling

Cooperative State University, Karlsruhe (CSU-K), Germany

nico.axtmann95@gmail.com, caro.mehret@gmail.com, berkling@dhbw-karlsruhe.de

Abstract

This paper presents the set-up and results regarding the Cooperative State University submitted system for the shared spoken CALL task. The data was collected from Swiss teenage students using a speech-enabled online tool for English conversation practice. The tasks consisted of training data of a German text prompt with the associated audio file containing an English language response by the students. Problems therefore consisted of recognizing children's speech with foreign accent, grammatical and vocabulary problems and a number of false starts due to language learning issues.

The task is to create software that will decide whether each response is appropriate (accept) or inappropriate (reject) in the context of the prompt. Results are reported through a single D-value that is computed specifically for this task.

The contribution of this paper is a detailed analysis of a variety of changes to the baseline system (D = 1.69) and the analysis of their contribution to the overall performance. The paper reports the official result (D = 3.21) on the shared task test files but also goes beyond the originally submitted system (D = 4.79).

Index Terms: CALL, speech recognition, ESL

1. Introduction to Shared Task

The work presented in this paper was performed in response to the shared task described in [1]. For the purpose of completion, we will briefly review the basic idea behind the CALL application under study; further details can be found in the above publication.

The exercise to be recognized and scored is of the type prompt-response, where the German-speaking student is prompted to either respond to a request or translate a request or sentence into L2, which is English. The automated system should ideally accept a correct response or reject a student response if faulty and offer relevant support or feedback. There are 565 types of prompts (given as text in German, preceded by a short animated clip in English), namely to make a statement or ask a question regarding a particular item. ((1) at the train station, (2) getting to know someone, (3) at the tube station, (4) at the hotel, (5) shopping for clothes (6) at the restaurant, (7) at the tourist information office, (8) asking/giving directions).

A wide range of answers is to be allowed in response, adding to the difficulty of giving automated feedback. Incorrect responses are due to incorrect vocabulary usage, incorrect grammar, or bad pronunciation and quality of recording. The shared task corpus has been annotated with correct transcription and a correct/incorrect tag regarding grammar, vocabulary, pronunciation and fluency.

In designing the automated system it is important to give accurate feedback of correctness without frustrating the student with false negative feedback or letting the student become overly confident by returning too many false positives.

The system's mistakes can be mitigated by recognizing its own mistakes with low confidence level or giving precise feedback regarding its diagnostic of inaccuracy. However, this goes beyond the scope of the present paper but a rule-based approach is able to support this future application.

The rest of the paper will describe a 2-way decision system that either rejects or accepts the student answer as correct. The baseline system is briefly reviewed in Section 2. Sections 3- 5 describe our additions and variations to the baseline system proposed by the shared task resulting in the CSU-K system. Section 6 evaluates the CSU-K system given the training and test data in the shared task. Finally, the paper concludes with insights from the task and proposes some future changes to the system.

2. Baseline System Description

2.1. Shared Task Corpus

The data for the shared task was collected in 15 school classes at 7 different schools in the German speaking areas during a series of experiments in 2014 and early 2015. To compare automated system performance, human annotators judge each interaction in order to determine whether or not the utterance should have been accepted by the system. The training corpus contains 5,000 utterances. The testing corpus contains 996 utterances. The selected responses were balanced across gender, age, proficiency and motivation. The data is challenging due to the recording environment in school and the ensuing background noises.

Examples of the data are given in Table 1.

Table 1: *Sample Data-Set. L=Language, M=Meaning, I=Incorrect, C=Correct.*

ID	Prompt	Transcription	L	M
4596	Frag: mein Steak durchgebraten	i would like my steak well down	I	C
3709	Sag: Ich habe 2 jüngere Brüder	i have two young brothers	I	I

2.2. Baseline System

A baseline system is provided for the shared task in the form of the speech recognizer and a language model.

2.2.1. Speech Recognition

Acoustic models, language models and scripts for Kaldi [2], a state-of-the-art open-source recognizer platform are provided as a starting point as described in [3]. The provided model is a triphone DNN-HMM model which has been trained on the

training part of WSJCAM0 [4] and the shared task training data. The DNN-HMM model is acquired by adding a softmax layer on to a pretrained DBN structure [5]. The language model is a backed off bigram model trained on the shared task training data.

2.2.2. Grammar

A grammar, automatically derived from the data collection, is provided in XML format for 564 possible prompts with a total of 11,776 possible responses.

2.2.3. Baseline Performance

The training data was split into 90% training and 10% development test data. Each student answer is scored as accept/don't accept at the semantic and syntactic level and compared against human annotated truth [6]. The baseline system reached an initial WER of 14.81% and a D-Value (Section 6) of 1.694.

3. Improving Speech Recognition

The speech recognizer was changed by adjusting the acoustic model and the language model.

3.1. Changing the Acoustic Model

The baseline DNN-HMM model with a WER of 14.81% was retrained by applying speaker independent transformations Linear Discriminant Analysis and Maximum Likelihood Linear Transformation (LDA+MLLT) on top of the triphone model. LDA+MLLT has shown improvements in recognizing children speech [7]. The model achieved a WER of 13.80%. In the next step Speaker Adaptive Training (SAT) [8] is applied on top of the previous model using Feature-Space Maximum Likelihood Linear Regression (fMLLR) [9] which achieved a WER of 13.41%.

In the next step we experimented with the number Gaussians and leaves in the phone decision tree and retrained the models described above. Results listed in Table 2 show that the best model with 13.05% WER is achieved with 2500/30000 (Leaves/Gausseans) using MLLT+LDA+SAT.

Table 2: *WER for different acoustic models.*

numLeaves/ numGauss	**Model**	**WER**
2000/10000	Baseline	14.81%
	+LDA+MLLT	13.80%
	+LDA+MLLT+SAT	13.41%
2500/30000	Baseline	13.59%
	+LDA+MLLT	13.12%
	+LDA+MLLT+SAT	13.05%

3.2. Changing the Language Model

The language model in the baseline system is a backed-off bigram model trained on the shared task training data. The following steps were used to improve the model.

1. To improve the language model, a new interpolated trigram model was trained with the provided shared task training data and the responses in the reference grammar. The SRILM toolkit [10] was used for this purpose.

Table 3: *WER after changing LM.*

numLeaves/ numGauss	**Model**	**WER**
2500/ 30000	Baseline + TRI	12.02%
	+LDA+MLLT+TRI	11.78%
	+LDA+MLLT+SAT+TRI	11.17%
	+LDA+MLLT+SAT+TRI+LM	10.72%

2. In the next step (+LM) we extended the language model by adding existing responses to the reference grammar with modified words. The following rules were applied in the following order:
   - I am → I'm (2274 additions)
   - a → one (10287 additions)
   - thanks → thank you (11954 additions)
   - I would → I'd (201 additions)
   - one → a (5776 additions)

Both changes are applied to the front end systems from Section 2 and the results are listed in Table 3, showing a general improvement, with the best model resulting in 10.72% WER.

3.3. Adapting the Pronunciation Dictionary

WER can be improved by adjusting the dictionary to include more expected pronunciations [11, 12, 13, 14]. We experimented with phonological rules to extend the pronunciation dictionary with different variants.

The following list shows the adjustments to the pronunciation dictionary according to typical German mispronunciations in English. After applying some of these rules [15] according to what we know about word-structure [16] to the dictionary, new entries were generated and added.

- dh (beginning) → d
- dh (beginning) → s
- v (beginning) → w
- dh (end) → th
- d (end) → t
- g (end) → k
- b (end) → p
- z (end) → s

However, as stated in the literature as well, children have limited linguistic knowledge and pronunciation skills. Therefore, the mismatch between regularity of the dictionary extensions and irregularity of pronunciations could not lead to improvements with this global, generative method but instead confused the recognizer further with too many variants. It seems that these adaptations to the user are better addressed in the acoustic space. No improvements but a higher WER resulted from these changes, so they were dismissed.

4. Preparing the Data

The shared task was supplied with a reference grammar. In order to use it well, several steps are taken to prepare a more robust grammar and to clean up the transcript from Section 3.

4.1. Pre-processing of Transcript

The transcribed utterance is first cleaned for further processing.

White space: All irregular white-space is removed and replaced with a single empty space.

Filler words: Superfluous words like "yes", "thanks", "thank you", "please" and "also" are removed as they have no influence on meaning and linguistic correctness. Some sentences starting with "no" and "and" fail when matching with the reference grammar. These words are removed as were words at the end of sentences (such as "no" and "is"), both of these may be artifacts resulting from erroneous parsing of noise.

Abbreviations: Recognized abbreviations are expanded. For example, "I'm" becomes "I am". A total of 9 such different abbreviations were changed in this manner.

Unique Words: Word duplication due to false starts or repetitions are difficult to match with a regular grammar. They are therefore removed during this pre-processing phase.

Typically Confused Words: Some words are very difficult for the speech recognition system. One such example is the word "desert" vs. "dessert". Similarly, "pm" is recognized by letters "p m". These types of words were manually mapped into their correct words, given what we know about the task (which did not talk about desert).

4.2. Extending the Reference Grammar

In this system, the reference grammar was adjusted slightly by adding a number of utterances in two steps.

1. Delta Grammar: Adding correct answers from the training data that did not appear in the original reference grammar.
2. By comparing the delta grammar with the original reference grammar new structures were derived. These are listed below.

In step two the following minor changes with major impact were addition of new sentences to the grammar that were generated through a number of word substitutions as follows: "one" $\rightarrow$ "a", "a" $\rightarrow$ "one", "want" $\rightarrow$ "need", "need" $\rightarrow$ "want", "this" $\rightarrow$ "that", "that" $\rightarrow$ "this", "o'clock" $\rightarrow$ "pm", "pm" $\rightarrow$ "o'clock", "night" $\rightarrow$ "evening", "evening" $\rightarrow$ "night", "for" $\rightarrow$ "to", "to" $\rightarrow$ "for", "these" $\rightarrow$ "those", "those" $\rightarrow$ "these".

4.3. Creating a POS-level Reference Grammar (submitted)

The originally submitted system (OLD) is described here. An index table was created for each prompt by looking at each of the answers at the POS (Part of Speech) level. An index table of vocabulary was created based on each word's POS within the sentence resulting in allowed words in correct answers by POS. Since the sentences are fairly easy, this simple approach is appropriate. An incoming utterance is presented in terms of POS.

4.4. Language Model Score (NEW - revised)

The revised system (NEW), after the deadline is described here. Using the SRILM Tool kit to train a language model on the augmented reference grammar resulted in a Trigram model that returns a sentence score of log probability. The score reflects the goodness of the sentence syntax, given the reference grammar. It can be used as a cut-off score for accepting syntax. Using Scikit[1], a decision tree was trained with the training and development test sets, resulting in a cut-off score (-29645.3398) for classification.

[1] http://scikit-learn.org/stable/

4.5. Creating Clusters of Prompts

A number of prompts are very similar. Therefore, some prompts are merged according to their similarity at the POS level, resulting in a total of five clusters of prompts (not all prompts belong to a cluster). From these clusters, words that carry the meaning are extracted in order to match these against the incoming utterances to these prompts.

The following lists these clusters and shows an example POS grammar for the first cluster. These are used to classify incoming prompts into one of the clusters.

Pay Cluster: This cluster combines POS structures for all prompts employing payment options. While syntax structure is given, key words differed by prompt. An example cluster is listed in Table 4 and clearly shows how tightly related these prompts are regarding the syntax structure of the prompt.

Table 4: *Prompts with similar POS structure (PRP MD VB TO VB IN NNS) for example Paying-Cluster:*

Pay Cluster
Sag: Ich möchte mit $Dollars$ bezahlen
Sag: Ich möchte mit $Euros$ bezahlen
Sag: Ich möchte mit $Kreditkarte$ bezahlen
Sag: Ich möchte mit $Mastercard$ bezahlen
Sag: Ich möchte mit $Postkarte$ bezahlen
Sag: Ich möchte mit $Visa$ bezahlen
Sag: Ich möchte mit $Pfund$ bezahlen
Sag: Ich möchte mit $SchweizerFranken$ bezahlen

Restaurant Cluster: This cluster combines POS structures for all prompts restaurant options like "Frag: Ich möchte die Rechnung", "Frag: Ich möchte die Dessertkarte".

Room Cluster: This cluster combines POS structures for all prompts hotel booking options.

Capital Cluster: This cluster contained all prompts asking for capitals and countries, such as "Sag: Die Hauptstadt von der Schweiz ist Bern". Hard-coded rules as to country and city that have to appear in the answer are applied.

Ticket Cluster: This cluster includes all prompts that ask for tickets for particular shows (Mamma Mia) or particular days (Monday night) and certain numbers.

5. Classification

The post-processed transcript uses the constructed reference grammar and clusters described in Section 4 through a series of rule-based expert modules for final classification of syntax and semantics for each utterance.

5.1. Basic Response Matching

Response matching is done at word and Part of Speech (POS) levels as described next. Details of each step will be explained subsequently.

1. If the cleaned transcript is matched by the augmented reference grammar (processed as in Section 4) then both syntax and semantics are classified as correct. Classification is finished.

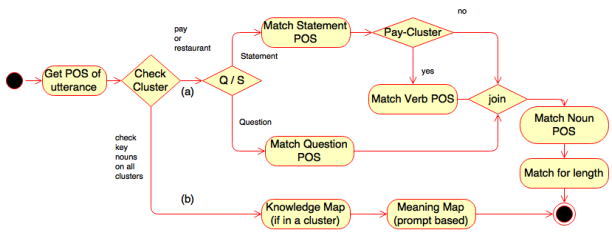

Figure 1: *Schematic diagram for (a) Clusters (Pay and Restaurant) using POS-level reference grammar (semantic, syntax) and (b) Cluster and Prompt matching for all (semantic).*

2. (NEW:) The sentence score (4.4) is applied to the incoming utterance. If the sentence score is below the threshold (negative log probability), ie. it is low enough to reflect a well-formed sentence given the reference grammar, the utterance is classified as semantically correct.
3. If the utterance belongs to prompts "Pay-Cluster" or "Restaurant-Cluster", then it is represented at the POS-level.
    - If the POS-structure matches within its cluster, the syntax is classified as correct (see Section 5.2).
    - (OLD:) Words in the sentence are indexed by POS and matched against those words indexed for this POS in the corresponding prompt. If the vocabulary matches, the semantics is classified as correct. (see Section 5.3)

While the first two steps are straight forward, the third one is explained below.

5.2. POS-Level Syntax Judgement

The utterance is attempted to be matched by a module, if it did not find a direct syntactic match in the reference grammar or pass the cutoff-sore in the second step (Section 4.4) and the utterance belongs to one of Pay- and Restaurant-Cluster.

This step works at the POS-level of the sentence. However, during the process of regenerating an extended reference grammar to improve coverage, a simplified approach was adapted. The idea of generating a reference grammar with extended coverage was applied to sub-problems only.

Looking at a subset of the known prompts, it was possible to determine syntactically correct sentence patterns for some prompts at the POS-level. This would allow some freedom for the speakers at the word level that is not covered by the reference grammar, thereby making the grammar more robust against out of vocabulary answers that may still be correct syntactically.

This approach was implemented for two most frequently occurring clusters, namely the Pay- and Restaurant-Cluster that make up 10-15% of the whole testing data. Other clusters (as described in Section 4.5) exist but did not get implemented yet.

Each POS-level Grammar contains the following components:

1. The POS-level reference Grammar distinguishes between the main categories of question vs. statement phrasing followed by
2. the POS-level content component and
3. the matching of length of incoming utterance with reference sentence.

Table 5: *Question Part of Speech and Examples*

Question POS	**Example**
VBZ PRP JJ TO VB	is it possible to pay
MD PRP VB IN	can you tell me
MD PRP VB	do you accept
VBP PRP VB	can I pay

Table 6: *Statement Part of Speech and Examples*

Statement POS	**Example**
PRP MD VB TO	I would like to
PRP MD VB DT	I would like a
PRP VBP TO	I wish to

This combination is shown in Figure 1, a) for their respective clusters. The details are explained below.

5.2.1. *Question vs. Statement*

Different POS patterns are used to extract questions as shown in Table 5. These were compiled from the data for each of the clusters.

Similarly, POS patterns are used to extract Statement patterns are shown in Table 6.

5.2.2. *Content*

Content is extracted by matching the content part of the sentence structure. These are usually verb and noun structures. Because the utterances in this application are so simple, this approach covers the sentence syntax in combination with the question/statement constructs described above.

Verb Constructs: Valid verb constructions are listed in Table 7.

Noun Constructs: Valid noun constructions are listed in Table 8.

Table 7: *Content POS Examples for Verb Structures*

Verb POS	**Example**
VBP DT	want a
VBP IN	pay by
VB DT	find the
VB IN	leave on

5.2.3. *Length*

Finally, after matching of substrings in both target grammar and incoming utterance, no extra words should remain. Having passed all stages, the utterance is judged to have correct syntax. The semantic judgement is described in the next section.

5.3. Rule-Based Meaning Judgement

Since we are looking for both meaning and syntax separately, this section discusses, how meaning can be judged as correct. In order to judge correct meaning apart from correct syntax, it is useful to extract certain POS features from the incoming utterance.

1. Match Nouns at POS level
2. Match words identified as nouns
    - Knowledge-based Matching of nouns

Table 8: *Content POS Examples for Noun Structures.*

Prepositions:	**Example**
IN, NN, NN	for friday night
IN, NN, NNS	with credit cards
IN, NN	for friday
IN, NNS	with dollars
Numbers:	**Example**
CD, NN	one ticket
CD, NNS	two tickets
Adjektives:	**Example**
JJ, NN	musical ticket
JJ, NNS	musical tickets
Determinants:	**Example**
DT, NN	the sweatshirt
DT, NN, NN	a grocery store
Personal pronouns:	**Example**
PRP$, NN	my room
PRP$, NN, NN	my master card

- Meaning Map (using Table generated in Section 4.3)

5.3.1. Key Nouns

Nouns are extracted by matching substrings at the POS-level in the incoming utterance as described in Table 8. The words found as nouns are then matched in two ways described below.

5.3.2. Knowledge-Based Matching of Nouns (by cluster)

The knowledge-based approach looks at nouns in the context of the cluster. If the expected nouns are found in the utterance, the meaning is judged as correct.

In the following clusters the noun is extracted from the utterance transcription and matched with the nouns in the responses defined by the reference grammar for the relevant prompt. The nouns are extracted according to the algorithm described in Section 5.1.

Ticket Cluster: The ticket cluster always asks for a number of tickets for a particular show or musical. The correct response must include the name of the show as well as the word for ticket. This cluster always asks for a certain number of tickets for a particular evening. For example, the Prompt "Frag: 2 Tickets für König der Löwen" necessarily contains the nouns "tickets" and "the lion king" or "lion king".

Capital Cluster: A number of prompts ask for capital cities of certain countries. A method of extracting both of these nouns and checking against a general knowledge base has shown to be helpful in identifying correctness of meaning. For example, the list of nouns required for the Prompt "Sag: Die Hauptstadt von der Schweiz ist Bern" necessarily must contain "switzerland" and "bern" and "capital".

Other: Pay, Restaurant and Room Cluster work in similar ways.

5.3.3. Meaning Map for Noun Matching (prompt based)

If a prompt was not matched with the above rules within their cluster, it uses a meaning map to match the utterance nouns against a list of nouns that were established for the corresponding prompt from the reference grammar. If there is a match, the meaning is judged to be correct.

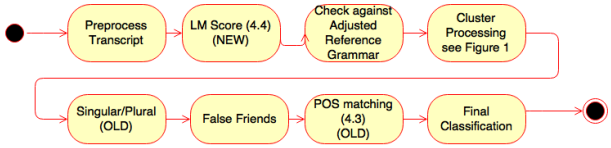

Figure 2: *Schematic diagram of rule-based post-processing pipeline.*

5.3.4. False Friends

A list of false friends that can be extended in future work was added to the system to identify false meaning. This list currently holds only the word "dessert card" for "dessert menu", which was frequently mistakenly used.

5.3.5. High-Frequency Errors

A number of high-frequency errors that students commit could be added as specific rules. For the present system, we implemented only one rule to detect incorrect usage of singular/plural. Future work can go into more detail here to extend coverage.

5.4. Pipeline-Based Final Classification

The overall system is depicted in Figure 2. Before entering the pipeline of modules, meaning and syntax are both set to false. As the utterance makes its way through the pipeline, intermediary decisions can be overwritten.

1. Pre-processing: The transcripts are formatted and syntax and semantics are set to FALSE (see Section 4.1).
2. Language Model: According to the score, the utterance syntax can be overwritten with TRUE (see Section 4.4).
3. Reference Grammar: A match with the augmented reference grammar can result in both syntax and semantics to be set to TRUE (see Section 4.2).
4. Cluster: According to clusters (as described in Section 4.5) both syntax and semantics to be set to TRUE for Pay and Restaurant Clusters and semantics can be set to TRUE for other clusters.
5. Meaning Map: Finally, the meaning map that works on each prompt (regardless of cluster) can be use to set semantics to TRUE.
6. Singular/Plural mismatch can reset the meaning to FALSE.
7. False friends match can reset the meaning to FALSE.
8. POS-level sentence matching was used in the system that was submitted to the competition and has since been replaced (see Section 4.3).

The final classification is written into the results file. If both syntax and semantics are TRUE the utterance is accepted, otherwise rejected.

6. Evaluation

6.1. Description of Test Data

Test data for the shared task was released two weeks prior to hand-in of the results. It consisted of the following items (similarly to the training data but without the transcriptions or judgements):

1. A set of 996 audio files
2. A CSV file of metadata, including ID, prompt and a wavefile. (11336 Frag: rote Stiefel 11336.wav)

6.2. D-Metric

The D-Metric given in Equation 1 is used to evaluate the system performance. The variables in the equation are defined as the number of utterances that fall into each of the following categories.

- **CR** Correct Reject, **CA** Correct Accept, **FR** False Reject
- **PFA** Plain False Accept (the student's answer is correct in meaning but incorrect English, the system accepts)
- **GFA** Gross False Accept (the student's answer is incorrect in meaning, the system accepts)

False Accept is defined by $FA = PFA + k.GFA$, where k, a weighting factor that makes gross false accepts relatively more important is set to 3.

$$D = \frac{(CR/(CR+FA))}{(FR/(FR+CA))} = \frac{CR(FR+CA)}{FR(CR+FA)} \quad (1)$$

6.3. Results

In this paper, we report on two results. Firstly, the system that was submitted to the competition at SLaTE 2017. In this system the OLD module described in Section 4.3 was used instead of the NEW module that was added later. In addition the acoustic model was retrained on the complete training set. Results are given in Table 9.

Table 9: *Results, where Pr=precision, R=recall, F=F-measure. (BK=Baseline Kaldi, OS=Our System, PPP=Best Submitted Of Our Team, JJJ=Best Transcript from competition.)*

Name	Pr	Rec	F	FR	CR	D
BK	0.957	0.439	0.602	0.951	0.561	1.694
BKOS	0.945	0.599	0.733	0.914	0.401	2.280
PPP	0.838	0.795	0.816	0.660	0.205	3.217
PPPOS	0.897	0.779	0.834	0.789	0.221	3.578
JJJ	0.871	0.848	0.859	0.717	0.152	4.710
JJJOS	0.872	0.839	0.856	0.723	0.161	4.503
NEW	0.903	0.835	0.868	0.791	0.165	4.799

7. Conclusions and Future Work

A rule-based system lends itself well for giving intelligent feedback to the learner. In this paper, we have attempted to build a rudimentary prototype of such a rule-based system. It is easier to understand where the student needs support, such as vocabulary or syntactic issues. A pipeline architecture allows us to separate meaning from syntax and hone in on problem areas. A lot more work is required to build a helpful feedback mechanism. Many of these rules are also very application dependent and may not generalize well to new problem sets. In future, as more data becomes available new approaches can be added to build hybrid systems. It is interesting to note that most of the system performance was gained by understanding the rule-based modules and using this to extend the reference grammar. The final system gains most of its leverage from the extended reference grammar. We expect the modules to support robustness against new data.

8. Acknowledgements

This work was performed by Bachelor students as part of their capstone project. The authors would like to thank Xizi Wei for visiting to help with Kaldi. We also thank the support of our "Förderverein".

9. References

[1] C. Baur, J. Gerlach, M. Rayner, M. Russell, and H. Strik, "A shared task for spoken call?" in *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*. Paris, France: European Language Resources Association (ELRA), May 2016.

[2] D. Povey, A. Ghoshal, G. Boulianne, N. Goel, M. Hannemann, Y. Qian, P. Schwarz, and G. Stemmer, "The kaldi speech recognition toolkit," in *In IEEE 2011 workshop*, 2011.

[3] M. Najafian, "Acoustic model selection for recognition of regional accented speech," Ph.D. dissertation, Ph. D. dissertation, University of Birmingham, 2016.

[4] T. Robinson, J. Fransen, D. Pye, J. Foote, and S. Renals, "Wsjcamo: a british english speech corpus for large vocabulary continuous speech recognition," in *Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995 International Conference on*, vol. 1. IEEE, 1995, pp. 81–84.

[5] G. Hinton, L. Deng, D. Yu, G. Dahl, A. rahman Mohamed, N. Jaitly, V. Vanhoucke, P. Nguyen, T. Sainath, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech recognition," 2016.

[6] C. Baur, "The potential of interactive speech-enabled call in the swiss education system: A large-scale experiment on the basis of english call-slt," Ph.D. dissertation, University of Geneva, 2015.

[7] D. Elenius and M. Blomberg, "Adaptation and normalization experiments in speech recognition for 4 to 8 year old children," in *Proc. Interspeech, 2749 – 2752*, 2005.

[8] T. Anastasakos, J. W. McDonough, R. M. Schwartz, and J. Makhoul, "A compact model for speaker-adaptive training," in *The 4th International Conference on Spoken Language Processing, Philadelphia, PA, USA, October 3-6, 1996*, 1996.

[9] M. J. F. Gales, "Maximum likelihood linear transformations for hmm-based speech recognition." *Computer Speech Language*, vol. 12, pp. 75–98, 1998.

[10] A. Stolcke, "Srilm – an extensible language modeling toolkit," in *International Conference on Spoken Language Processig (ICSLP)*. ISCA, 2002, pp. 901–904.

[11] P. Cosi, M. Nicolao, G. Paci, G. Sommavilla, and F. Tesser, "Comparing open source asr toolkits on italian children speech." in *WOCCI*, 2014, pp. 1–6.

[12] Q. Li and M. J. Russell, "Why is automatic recognition of children's speech difficult?" in *Seventh European Conference on Speech Communication and Technology*, 2001.

[13] M. Wester, "Pronunciation modeling for asr–knowledge-based and data-derived methods," *Computer Speech & Language*, vol. 17, no. 1, pp. 69–85, 2003.

[14] P. G. Shivakumar, A. Potamianos, S. Lee, and S. Narayanan, "Improving speech recognition for children using acoustic adaptation and pronunciation modeling." in *WOCCI*, 2014, pp. 15–19.

[15] E. Atwell, P. Howarth, and D. Souter, "The isle corpus: Italian and german spoken learner's english," *ICAME Journal: International Computer Archive of Modern and Medieval English Journal*, vol. 27, pp. 5–18, 2003.

[16] K. Berkling, "Scope, syllable core and periphery evaluation: Automatic syllabification and foreign accent identification," *Speech Communication*, vol. 35, no. 1, pp. 125–138, 2001.