

# Language-agnostic Code-Switching in Sequence-To-Sequence Speech Recognition

Enes Yavuz Ugan<sup>1</sup>, Christian Huber<sup>1</sup>, Juan Hussain<sup>1</sup> and Alexander Waibel<sup>1,2</sup>

**Abstract** Code-Switching (CS) is referred to the phenomenon of alternately using words and phrases from different languages. While today’s neural end-to-end (E2E) models deliver state-of-the-art performances on the task of automatic speech recognition (ASR) it is commonly known that these systems are very data-intensive. However, there is only a few transcribed and aligned CS speech available. To overcome this problem and train multilingual systems which can transcribe CS speech, we propose a simple yet effective data augmentation in which audio and corresponding labels of different source languages are concatenated. By using this training data, our E2E model improves on transcribing CS speech. It also surpasses monolingual models on monolingual tests. The results show that this augmentation technique can even improve the model’s performance on inter-sentential language switches not seen during training by 5,03% WER.

## 1 Introduction

Due to increasing globalization, a growing number of people move to foreign countries to make a living an example would be Germany which shows an increase from 9.107.895 foreign population in 2015 to 11.817.790 in 2021 [9]. As these people start learning a new language, this can result in Code-Switching (CS), which is referred to as the change between languages while speaking. An example of German-English CS would be the phrase ‘Das war sehr strange’ (‘That was very strange’).

From a linguistic perspective, CS can be divided into multiple categories [37]:

- Inter-sentential CS: The switch between languages happens at sentence boundaries.

---

<sup>1</sup>Interactive Systems Lab, Karlsruhe Institute of Technology, Karlsruhe, Germany

<sup>2</sup>Carnegie Mellon University, Pittsburgh PA, USA

firstname.lastname@kit.edu, alexander.waibel@cmu.edu

- Intra-sentential CS: Here the second language is included in the middle of the sentence. Additionally, the word borrowed from the second language can happen to be adapted to the grammar of the matrix language as well.
- Extra-sentential CS: In this case, a tag element from a second language is included, for example at the end of a sentence. This word is more excluded from the main language.

As these developments can result in growing numbers of multilingual communities and individuals the need for dialogue and ASR systems capable of processing such CS data is very important. Despite occurring frequently, CS poses a great challenge for all neural-network-based end-to-end ASR models. As of today, there are only a few CS data available for a very limited number of languages. Some example corpora available are [3] for CS between French and Algerian speech, [29] containing utterances switching between Mandarin and English, and [6] having gathered data with CS between English and Cantonese.

As an exemplary case, in this work, we focus on developing a multilingual ASR system capable of transcribing CS utterances between German and English. Considering the increased amount of Arabic-speaking people in Germany another common language that is mixed with German is Arabic and thus we included it as a third language. These languages are specifically interesting to analyze as German and English are from the same Indo-European language family while Arabic is part of the Afro-Asiatic language family. As training data is not available in our scenario, we conduct multiple experiments using a straightforward CS data augmentation technique. Specifically, we present the following contributions:

First, we present a simple yet effective data augmentation technique designed for CS models in data-scarce scenarios, by simply concatenating multilingual sources and corresponding targets without any language information. Second, we perform an extensive evaluation of our model on intra- & inter-sentential CS test sets, as well as monolingual ones. Next to using publicly available test sets, we conduct our evaluation on artificially generated as well as our in-house collected data. Our experiments yield interesting results section 6, including 1) enabling the Sequence-to-Sequence (S2S) model to reliably transcribe CS utterances, 2) improving the performance of the multilingual model on monolingual test sets and 3) the capability of transcribing CS utterances for language pairs not switched during training.

## 2 Related Work

As transcribing Code-Switching utterances inherently needs an ASR model which is multilingual to some degree, we want to refer to some of the early work in this research area such as [46], [41], [42], [32], [49] and [31].

As there are only a few CS data available there has not been too much research for many language pairs especially if there is no data present. Some of the language pairs addressed are Frisian-Dutch [51], Malay-English [1], dialectal Arabic-

English [17], different Indian languages with English in [11], Korean-English [26], Japanese-English [33],[34], and Mandarin-English [50], [28], [53], [7].

Most of the work on Code-Switching focuses on language pairs with some available CS data. In [45] the authors aim at solving the problem of code-switching using a multi-task learning (MTL) approach. The authors investigate training a model predicting a sequence of labels as well as predicting language identifiers at different levels of the architecture. They also report that first training with monolingual data and fine-tuning it with CS speech improves their performance. In [44] the writers analyzed the effect of fine-tuning toward CS data on monolingual ASR. They show that fine-tuning a model on CS and monolingual data yields a better overall Word Error Rate (WER)s than when only using Code-Switching data. In [27] the authors propose to train two separate models. One CTC model for speech recognition and another one for frame-level language prediction. During decoding, if the current frame has a very high probability for the blank symbol the blank label is emitted, otherwise the output probabilities of English tokens are multiplied by the probability of this frame being English and the Chinese labels are multiplied by the probability of this frame being Chinese speech. While improving the CS WER they report a decrease in monolingual speech recognition performance. To improve the model's performance on CS speech in [54] the authors propose language-related attention mechanisms to profit more from using monolingual data, next to CS ones.

Other works try utilizing CS training data in order to use it for data augmentation. In that way, they aim at improving the performance by utilizing more than the original available CS data. In [52] the authors used a separate TDNN-LSTM [36] as an acoustic model, as well as a separate language model. Thus they were able to utilize CS speech-only data for enhancing the acoustic model. They also enhanced their language model separately using artificially created text-only CS data. Thus they were able to improve over a baseline model only trained with the original CS data. Another work [16], is also using a semi-supervised approach focusing on improving the lexicon and the acoustic model of an HMM-based ASR model. First, they extend the lexicon to realize no out-of-vocabulary for their training data. As in their CS data, English words have accented pronunciations, afterwards, they use a phonetic level decoding to learn adapted pronunciations of words. Additionally, they used audio of transcriptions with high Word Matched Error Rate in order to improve their acoustic model in a semi-supervised fashion. Each of their steps yields improvements in the English-Chinese CS setup on which they evaluated. In [12] the authors propose three different data augmentation algorithms. They apply audio splicing, meaning they randomly insert audio segments of the same speaker in a different language into the original utterance. The other two approaches are randomly inserting or translating a word in the source text and generating the corresponding audio using a TTS system. Here, the TTS system is trained with CS data, and also the word alignments needed for audio splicing were retrieved using an HMM-GMM ASR system trained on the initial CS data.

However, some of the earlier work also considered developing ASR models without the use of transcribed CS data. In [43] the authors train a hybrid attention/connectionst temporal classification (CTC) network which first classifies which

language is going to be transcribed followed by the transcription itself. In [34] the authors address the task of CS in ASR and TTS using a semi-supervised learning approach using the machine speech chain. In their approach first, an ASR and a TTS model are trained separately using monolingual data. Afterward, they utilize speech-only data by first transcribing it and re-synthesize the transcription in order to update the TTS model. CS text-only data is utilized by synthesizing the transcript and then transcribing it afterward. That way the ASR model gets trained for the CS task. The authors also use speaker embeddings in order to make sure the synthesized speech is the same speaker as is in the input. Their strategy improves CS performance without using any transcribed CS data but improves even further if some paired CS data is used as well. Another interesting data-augmentation approach is presented in [20]. They propose an approach consisting of multiple steps, in order to generate artificial CS text-only data. First, some Arabic text is translated into English. The Arabic script is morphologically segmented in order to calculate a better alignment between the translations. Afterward, a sentence-level constituent parse tree is generated and the CS data is generated according to the Equivalence Constraint theory described in [38]. This data is used to improve the Language Model of their HMM-GMM ASR model.

As can be read, most of the previous work considers cases in which some CS training data is available. We were interested in training a S2S model, without any real CS data, which is the more common case considering the data available. Our model should not be explicitly trained to predict languages but should do so implicitly by predicting the right labels, which in our case are Byte pair encoding tokens. Additionally, we analyzed the effects of different data augmentation constraints on the model’s performance.

### 3 MODEL

For our experiments, we used a S2S encoder-decoder-based model, as described in [35]. An abstract description of the neural network would look like this:

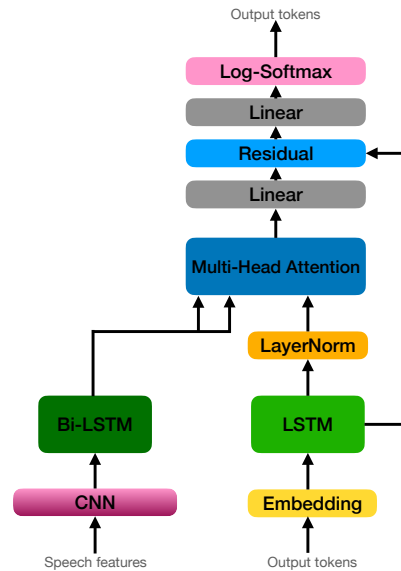
$$\begin{aligned}
 enc &= Bi-LSTM(CNN(logMel\_Spectrum)) \\
 tgt\_emb &= LSTM(Embedding(out\_tokens)) \\
 dec &= (MHA(enc, enc, tgt\_emb) + tgt\_emb) \\
 output &= log\_softmax(dec)
 \end{aligned}$$

In more detail, the model consists of a two-layer Convolutional Neural Network (CNN) applying 32 channels. We choose the window size of three over the frequency, as well as the time domain. A stride of two was applied resulting in a spectrogram down-sampled by a factor of four. After down-sampling a six-layer bidirectional LSTM is adopted. The decoder consists of an embedding layer followed by a two-layer unidirectional LSTM. The hidden size for all LSTMs was set to 1024. The output of the encoder and decoder LSTMs are used to calculate a context vector

using a multi-head cross-attention mechanism [47] with eight heads. After applying a residual connection with the decoder LSTM output, a projection layer is used to project the hidden dimension size to the size of the vocabulary. The architecture is depicted in Fig. 1.

As input, we use 40-dimensional log-Mel features calculated on frames of 25 ms with a stride of 15 ms. In contrast to some of the previous other works, we use one Byte pair encoding (BPE) [13] calculated on all three languages. This means we have in total 4000 labels for all languages. When calculating the BPE we made sure to use the same amount of text data for the three languages. The resulting BPE contains 2553 English, German, and 1444 Arabic tokens. The other three tokens are the unknown, start of sequence, and end of sequence tokens. The labels for the monolingual experiments were calculated on monolingual text data. We decided to use BPE tokens as they have shown to yield good results in the ASR task. Another reason we did not choose some common representation space for the Latin alphabet and the Arabic abjad is that we aim at printing the transcription in the correct language without any additional systems needed. If we transliterated Arabic into Latin script the question would arise if the hypothesis is actually an Arabic or English/German transcription of the speech.

The same number of model parameters are used in all our experiments. 1024 dimensional LSTMs are trained using Adam optimizer [23] with a maximum learning rate of 0,002 and 8000 warm-up steps. After each epoch, the perplexity is used to determine if the model improved or decreased in performance. The validation performance was determined by adding the perplexity of the monolingual validation sets of each language, as well as a pseudo-CS validation set generated using the



**Fig. 1** Abstract architecture of the encoder-decoder-based Sequence-to-Sequence Model used in our experiments.

same three validation sets by applying the algorithm explained in section 5. An early abortion was applied if there were no improvements over five epochs. For tests, the epoch with the lowest validation perplexity during training was chosen.

## 4 DATA

As already mentioned in Section 1 we used three languages in this work, namely Arabic, German and English. The English training data is made up of How2 [40] and TED-LIUM (TED) [39] data sets. For the German training data we used Common Voice (CV) [4], Europarl [24], Lect. a data set of recorded lectures and interviews, and Mini-international Neuropsychiatric Interview (MINI)-Data. As Arabic training data, we used MGB2 (Alj.) data from [2] and MINI data [19]. An overview of our training data is given in Table 1.

**Table 1** Data used during training.

Language	Corpus	Speech [h]	Utterances
English (EN)	How2	345	210k
	TED	439	259k
German (DE)	CV	314	196k
	Europarl	46	20k
	Lect.	504	353k
	MINI-Data	1	498
Arabic (AR)	Alj.	1127	375k
	MINI-Data	39	9k

For our tests, we have monolingual test sets for each language. The Alj.2h data was dialect-free Arabic data extracted as explained in [21]. In order to evaluate the CS performance, we generated a test set (artificial) by applying the data augmentation technique, using CV, Alj.2h, and WSJ test sets, as described in Section 5. For intra-sentential CS with German as the matrix and English as the embedded language, we use our in-house Lect. test set where English words have been manually annotated. We combined this data with a small set of read speech, collected by us. This is depicted as Deng. in Table 2. Here the overwhelming amounts are German words and only 4,5% are English. We also tested our models on the German-English intra-sentential CS test set derived from the Spoken Wikipedia Corpus (SWC) provided by [22], depicted as SWC-CS. Detailed information about our test sets can be taken from Table 2. Both of these intra-sentential sets have German as the matrix language with English words embedded. tst-inter is an inter-sentential CS set we derived from MuST-C (tst-COMMON) [10] data. At sentence boundaries, the sentence was continued in either English or German in a CS manner. These sentences were then read by two persons. We also collected a German-English CS test set (D-E-CS) and a German-Arabic test set (D-A-CS) which contain switches at depen-

dent and independent clauses and as such contain longer intra-sentential CS data, as well as inter-sentential Data. This data was generated by using our Lect. test set and tst-common and translating clauses into the respective language. Afterward, the utterances were read by 4 and 2 speakers for the D-E-CS and D-A-CS respectively, using the TEQST tool <sup>1</sup>. In D-E-CS 57,3% of the clauses are German and 42,7% are English. In D-A-CS 50% of the clauses are German and 50% are Arabic. While the speakers reading in the D-E-CS set were of German origin, the speakers reading D-A-CS originated from Arabic countries. Participants reading utterances containing English were L1 in German and L2/B1 in English. Speakers recording text with Arabic as a language pair were L1 in Arabic and L3/B1 in German.

**Table 2** Data used for testing.

Language	Corpus	Speech [h]	Utterances
English	TED	3	1k
German	CV	25	15k
	Lect.	5,2	5k
Arabic	Alj.	10	5k
	Alj.2h	2	1k
Intra-sent.	SWC-CS [22]	34,1	12437
	Deng.	0,95	293
Inter-sent.	artificial	1,9	1687
	tst-inter	0,85	284
Mix-sent.	D-E-CS	1,42	562
	D-A-CS	1,09	398

## 5 APPROACH

Inspired by [43], we applied a concatenation technique to generate our CS data. We concatenate the log-Mel features of different languages after each other. For the target labels, we also concatenate the respective labels after each other. We want to note, that the speakers are not the same in each language, as such the resulting data can not be considered CS data but more like pseudo-CS data. As it turned out to be an important factor in training the model, we enable to set a specified relative amount of CS utterances in the training set. During data augmentation, we only have two restrictions. First, we limit the amount of CS data to a specific percentage. As we need to define a restriction on how long concatenated utterances are allowed to be, we analyzed our monolingual training data. We saw that most utterances are less than 10 seconds long. Thus, our second restriction is that we limit the length of the CS data. 25% of the CS data was made to be five seconds, another 25% up to 10 seconds another 25% 15 seconds long. Utterances of 20 and 25 seconds each made

<sup>1</sup> <https://github.com/teqst>

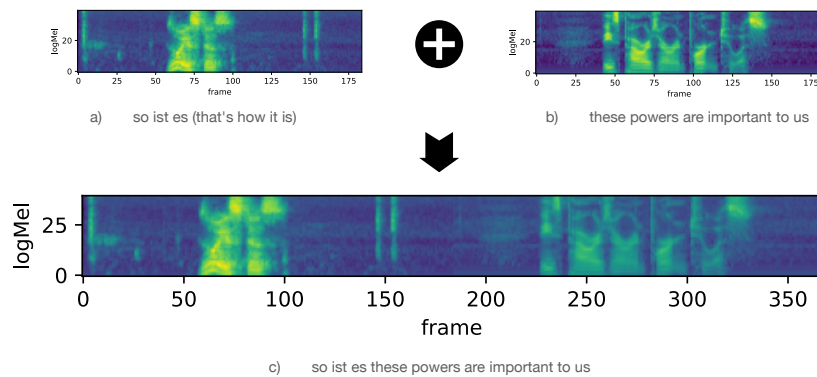
up 12,5% of the newly generated CS data. For each of the above-mentioned time ranges, we generate CS data the following way. First, a language is chosen randomly with an equally distributed probability. Afterward, an utterance is randomly picked out of all the sequences in that language. These steps are repeated until the CS duration of the sequence is up to two seconds shorter than the current time range. As we can see our data augmentation does not add any information about the language being transcribed and has no major restrictions. Keeping the process simple benefits an easier and more general usage of this approach. As, in intra-sentential CS cases, a word of the embedded language can be adapted to the grammar of the matrix language we believe that not predicting the language explicitly during decoding is also beneficial for training the model in a more general way.

An example input feature is shown in the following Fig. 2. In a) a monolingual German utterance which was randomly picked as described above is depicted. In b) a second utterance, this time a monolingual English one was selected. The algorithm we propose now concatenates their logarithmic Mel features as well as their transcript and passes them as the model input and the ground truth for teacher forcing.

## 6 EXPERIMENTS

### 6.1 Baselines

As for baselines, we trained four different models. One monolingual model for each of the languages Arabic (Mono-Ar), German (Mono-De), and English (Mono-En). The fourth baseline is a multilingual model (Mult.), which we trained using the



**Fig. 2** a) Monolingual German utterance. b) Monolingual English utterance. c) The concatenated features resulting in our pseudo-CS data for the target transcript "so ist es these powers are important to us".



concatenated data set of the three languages. In Table 3 the WER performances of these models are reported on monolingual test sets. While the multilingual model can transcribe all languages, a drop in performance can be seen in all tests when compared with the monolingual counterpart.

The performance of our baseline models on multilingual CS data is provided in Table 4. For Mono-Ar and Mono-En, it can be seen that the performance on CS data is quite bad. On our intra-sentential tests, the monolingual German model has the best results, this is due to German being the matrix language and as mentioned in 4, English words are only embedded sporadically in these utterances. Looking at tst-inter and the D-E-CS, it is visible that Mono-De and Mono-En perform very poorly as well. Mono-De has a slightly lower WER probably because there are more German clauses in the test set than English ones.

While the multilingual model decreased the performance by relative 7,69% WER on the intra-sentential CS, it was able to outperform the Mono-De model by relative 52,58% WER on tst-inter and relative 43,67% WER on D-E-CS.

Interestingly on D-A-CS, we can see similar scores for Mono-DE and Mult.. Looking at the transcripts we see that the multilingual model only transcribes parts of the utterance in one of the two languages. Similar to Mono-DE which only transcribes German parts of the utterance. Compared to the improvements in D-E-CS this shows that sharing the language scripts can have major benefits for multilingual models.

**Table 3** Results of baseline models on monolingual test sets. Results are reported in WER%.

	EN		AR		DE	
<b>model</b>	Ted (EN)	Alj.2h (AR)	Alj. (AR)	CV (DE)	Lect. (DE)	
Mono-De	-	-	-	<b>11,82</b>	<b>17,78</b>	
Mono-Ar	-	<b>9,74</b>	<b>16,00</b>	-	-	
Mono-En	<b>7,58</b>	-	-	-	-	
Mult.	9,25	10,48	16,64	17,15	21,27	

**Table 4** Results of baseline models on multilingual CS test sets. Results are reported in WER%.

<b>model</b>	DE-EN			DE-AR	DE-AR-EN	
	intra-sent.	inter-sent.	mix-CS	mix-CS	inter-sent	
	Deng.	SWC-CS	tst-inter	D-E-CS	D-A-CS	artificial
Mono-De	<b>18,99</b>	<b>28,97</b>	48,55	50,83	64,00	83,87
Mono-Ar	101,03	110,89	100,32	101,36	70,04	73,85
Mono-En	104,60	118,83	63,47	67,11	117,14	84,89
Mult.	20,45	31,19	<b>23,02</b>	<b>28,63</b>	<b>60,74</b>	<b>39,88</b>

## 6.2 Data augmented Code-Switching

In our first experiment, we trained a multilingual model using the data augmentation described in Section 5. Directly training the model with 50% artificially created CS data leads to a bit more unstable gradients. We trained the model multiple times. While the performances were not that different the number of epochs needed for training was very different 109 and 198 for Mult.-noc1 and Mult.-noc2 Table 5 respectively. We reason the unstable gradients to be present due to the difficult data, as well as the nature of the task. While the Arabic language is Phonetically and script-wise very different from German or English, the quality of the used audio can also increase the difficulty of the task.

As mentioned in [5] we apply a curriculum learning and first train on monolingual data, which can act as a regularization. For the second stage of the curriculum, we took the multilingual model from Section 6.1 and used the epoch with the lowest perplexity as a pre-trained model. The same training hyper-parameters are applied as in the first training of the model and all weights are updated. This model is denoted as Mult.cur50 and was trained with 50% CS augmented data. This model was trained in only 39 Epochs compared to 109 Epochs without curriculum learning which shows a significantly faster convergence. We also applied the second curriculum step with only 20% CS augmented data to see the effect it has on the training (Mult.-cur20). As we have more updates with a higher learning rate in the two-stage approach we also trained the initial multilingual model a second time without CS data (Mult.-noCS).

The results of monolingual tests are shown in Table 5. As Mult.-cur20 has a slightly better performance compared to Mult.-cur50, we will focus on the model which was trained with 20% CS augmented data. We can see that training the CS models with the two-step approach yield the best performances and even outperform the monolingual models in Table 3. The only exception is the German CV test, however, while the Mult.-noCS model has a relative decrease of 17,93% WER, training with CS mitigates the drop in performance to only a 9,64% decrease compared to Mono-DE.

**Table 5** Results of multilingual models on monolingual test sets. Results are reported in WER%.

model	EN		AR		DE	
	Ted (EN)	Alj.2h (AR)	Alj. (AR)	CV (DE)	Lect. (DE)	
Mult.	9,25	10,48	16,64	17,15	21,27	
Mult.-noCS	7,76	10,22	15,44	13,94	17,84	
Mult.-noc1	7,77	9,84	15,82	14,93	18,68	
Mult.-noc2	8,67	10,70	16,82	16,76	20,87	
Mult.-cur50	<b>7,12</b>	9,32	<b>15,11</b>	13,23	17,82	
Mult.-cur20	7,14	<b>9,30</b>	15,33	<b>12,96</b>	<b>17,32</b>	

In Table 6 the CS results after the second-curriculum are depicted. On our own small intra-sentential Denglish set we see that training without curriculum (Mult.-noc1) hurts the performance, compared to the Mult.-noCS model which was trained without data augmentation. The other data-augmented models can roughly keep the same WER. On the bigger German-English intra-sentential SWC-CS test set we can observe a relative improvement of 2,27% and 2,58% WER for the Mult.-cur20 and Mult.-cur50 models over the baseline multilingual model (Mult.-noCS). More importantly, however, compared to training without CS data, utilizing a CS augmentation of 20% yields relative improvements of 10,76% WER on the tst-inter data and a relative improvement of 8,55% WER on the D-E-CS test set, as well as a relative improvement of 25,26% on D-A-CS. Similar to previous work we also evaluated our models on artificially created CS data with switches between all languages (DE-AR-EN). The Mult.-cur20 yields a relative improvement of 80,35% WER compared to the multilingual model without CS (Mult.-noCS), which is extremely high when compared to our collected in-house test data. This is why we ignore this test case in our ablation studies, as we believe that testing on artificial data yields inflating improvements, which do not hold on our collected data, although it is only read speech. Fig. 3 shows an example output on the tst-inter test set. This example shows that the

**Table 6** Results of multilingual models on CS test sets. Results are reported in WER%.

model	DE-EN				DE-AR	DE-AR-EN
	intra-sent.		inter-sent.	mix-CS	mix-CS	inter-sent
	Deng.	SWC-CS	tst-inter	D-E-CS	D-A-CS	artificial
Mult.	20,45	31,19	23,02	28,63	60,74	39,88
Mult.-noCS	16,38	28,64	20,91	25,98	53,90	44,32
Mult.-noc1	18,28	28,98	20,12	25,47	55,57	9,25
Mult.-noc2	19,30	31,24	19,82	26,79	54,97	10,73
Mult.-cur50	<b>16,23</b>	27,99	18,81	<b>23,63</b>	45,67	<b>8,66</b>
Mult.-cur20	16,40	<b>27,90</b>	<b>18,66</b>	23,76	<b>45,40</b>	8,71

model is now more reliable when it comes to switching the language when transcribing at the switching region. The difficulty of such transcription may also arise due to people speaking English with an accent of their first language. In general, the model trained with pseudo-CS data seems more reliable when transcribing words at switching points in the utterance.

The results depicted in Table 5 and Table 6 show that using CS augmented data does not just improve models on CS data but also improves the model’s performance over the monolingual model on the respective monolingual test sets, as well.

### 6.3 Ablation studies

After seeing the results in Section 6.2 we further analyzed the effect of utilizing this kind of artificially created pseudo-CS data during training. Specifically, in a scenario with many more languages, the question will arise if we need to ensure generating CS data with transitions between all possible languages, and do we need to ensure that bidirectional transitions from one language to all others need to be present to enable CS during inference?

For this reason, we conducted several further experiments. All experiments apply the same curriculum learning regime and use the multilingual model described in Section 6.1 as a starting point. The results are depicted in Table 7 and Table 8. The ending of model names depicts which transitions were not seen during training, for example, "nodear" means there was no transition from German to Arabic. "nodex" means that German utterances were not used in any CS data. In contrast, "odex" depicts the case, in which only transitions from and to German were present.

The results of monolingual tests Table 7 give interesting insights into using artificially created CS data for multilingual models. We can see that all models which saw CS data during training outperform the baseline multilingual model (Mult.-noCS). We can appreciate that usually depending on which language or language transition was kept out of the training the performance on the respective test seems to degrade slightly compared to the Mult.-cur20 model. The reason is that these restrictions make the other languages proportionally more present in the training data. This is also supported by the WER improvements on the other languages which were not restricted. An example would be the performance of Mult.-nodeen (third row) on the TED performance and the Alj.2h set. Compared to Mult.-cur20 (second row) the WER on Ted decreased from 7,14% WER to 7,21%, while the performance on Alj.2h improved from 9,30% WER to 9,12%.

Reference	jetzt stellt sich heraus that even if you do choose to participate wenn mehr möglichkeiten zur auswahl standen even then it has negative consequences
Mult.	jetzt stellt sich heraus <b>class</b> even if you do choose to participate wenn mehr möglichkeiten zur auswahl standen <b>evenden</b> it has negative consequences
Mult.-noCS	jetzt stellt sich heraus <b>dass</b> even if you do choose to participate wenn mehr möglichkeiten zur auswahl standen <b>ebendin</b> it has negative consequences
Mult.-cur20	jetzt stellt sich heraus that even if you do choose to participate wenn mehr möglichkeiten zur auswahl standen even then it has negative consequences

**Fig. 3** Transcription hypothesis for the Referenz: "jetzt stellt sich heraus that even if you do choose to participate wenn mehr möglichkeiten zur auswahl standen even then it has negative consequences" German parts are (now it turns out) (when there are more choices present)

**Table 7** Results of multilingual models on monolingual test sets. Models were trained using 20% data augmentation with varying restrictions. Results are reported in WER%.

model	EN	AR		DE	
	Ted (EN)	Alj.2h (AR)	Alj. (AR)	CV (DE)	Lect. (DE)
Mult.-noCS	7,76	10,22	15,44	13,94	17,84
Mult.-cur20	<b>7,14</b>	9,30	15,33	12,96	17,32
Mult.-nodeen	7,21	<b>9,12</b>	15,08	13,08	17,68
Mult.-nodear	7,19	9,23	15,15	12,84	17,36
Mult.-nodex	7,23	9,26	15,17	13,39	17,80
Mult.-odex	7,20	9,43	<b>14,90</b>	12,88	17,20
Mult.-noende	7,28	9,13	15,12	13,11	17,42
Mult.-noenar	7,22	9,14	15,06	12,86	16,99
Mult.-noenx	7,32	9,36	15,06	12,86	17,46
Mult.-oenx	7,28	9,42	15,34	12,91	17,42
Mult.-noarde	7,29	9,14	15,00	12,98	17,58
Mult.-noaren	7,20	9,44	15,52	12,89	17,31
Mult.-noarx	7,17	9,57	15,18	<b>12,68</b>	<b>16,92</b>
Mult.-oarx	7,19	9,01	15,06	12,97	17,00

Looking at intra-sent. evaluations, specifically the Deng. test, Table 8, we can observe similar behaviour to Table 7. In the second last row x-noarx the model trained with only German and English switches is depicted. Compared to x-cur20 (second row), this results in more German and English utterances being seen during training and thus slightly improves over the model by 3,48% relative WER. This is due to the intra-sent. test set only containing examples with German as the matrix language and English words embedded.

Another interesting point for Intra-sentential CS is the phenomenon, in which words from the embedded language, in our case English, can happen to be adapted according to the grammar of the matrix language. This being the case we also report the accuracy of correctly transcribed English words in our Deng. test set and report them in the Column (Deng.Acc) in Table 8. From these numbers, we can see that there is an inverse correlation between correctly transcribing English words and the overall WER on Deng. test data. However, as the data augmentation only uses CS between longer clauses we see that there is only a limited effect on such intra-sentential data.

When looking at inter-sent. and mix-CS examples for DE-EN language pairs, we can see that the model trained without German switches, x-nodex (fifth row), decreases performance compared to x-cur20. The performance on tst-inter decreased from 18,66% WER to 20,38%. However, it still performs slightly better than the baseline multilingual (x-noCS) model with 20,91% WER. Models which never saw switches from German to English, x-nodeen (third row), also lose a bit of performance compared to x-cs20. However, looking at the transcriptions we can see that the model is still able to transcribe switches from German to English, which shows that the model is able to generalize the possibility of switching between languages and not just learns one specific switch. The answer to one of the questions of this

ablation study is very well answered on the D-A-CS test data. Here the worst performing model which has seen any kind of Arabic CS data during training is the x-oenx model (tenth row) which only saw switches from and to English. On the D-A-CS test which only contained Arabic and German CS data, the performance of the model improves over the baseline multilingual model relative by 9,33% WER although the model never saw switches between aforementioned languages. This shows that when training models to transcribe CS, especially in the inter-sentential case there is no need to provide switches between all language pairs.

Considering the mentioned results, we can appreciate that the model generally benefits from seeing CS data during training. It not just improves monolingual performance but also improves on inter-sentential CS data. We can also see that the model has the general capability to learn to switch between languages never seen in a CS scenario during training. However, at least seeing the language switched one time with any other language greatly improves over not switching at all. Using each language at least once in any switching combination can massively improve the capability of the model in general, no matter if a specific language switch was seen during training or not.

**Table 8** Results of multilingual models on CS test sets. Models were trained using 20% data augmentation with varying restrictions. Results are reported in WER%. Deng.Acc is the accuracy of correctly transcribed English words in percentage.

Mult.	intra-sent.		DE-EN		DE-AR	
	Deng.	Deng.Acc	inter-sent.	mix-CS	mix-CS	
			SWC-CS	tst-inter	D-E-CS	D-A-CS
x-noCS	16,38	79,03	28,64	20,91	25,98	53,90
x-cur20	16,40	79,53	27,90	18,66	23,76	45,40
x-nodeen	16,23	79,87	28,00	19,36	25,02	43,81
x-nodear	16,34	80,87	27,92	18,19	23,30	46,85
x-nodex	16,95	79,19	28,47	20,38	25,63	51,26
x-odex	16,06	81,20	27,89	<b>17,39</b>	23,92	44,36
x-noende	17,30	79,03	27,99	18,58	24,33	43,18
x-noenar	16,18	80,03	<b>27,71</b>	18,61	24,55	46,22
x-noenx	16,19	79,87	27,98	21,58	27,29	41,84
x-oenx	16,39	80,87	27,99	18,21	23,85	48,87
x-noarde	16,38	80,20	28,09	18,08	<b>23,21</b>	46,96
x-noaren	16,23	80,54	27,95	18,53	24,10	45,04
x-noarx	<b>15,83</b>	<b>81,71</b>	27,80	18,00	24,43	55,77
x-oarx	16,54	80,37	28,10	19,66	25,25	<b>41,05</b>

### 6.3.1 Transformer architecture

In order to see the if this data-augmentation is generalisable we trained a Transformer based S2S model, as well. The model consists of two CNN layers and six encoder and four decoder layers with a hidden size of 1024.

**Table 9** Results of multilingual Transformer models on CS test sets. Results are reported in WER%.

Mult.	DE-EN			DE-AR	
	intra-sent.	SWC-CS	inter-sent.	mix-CS	mix-CS
	Deng.		tst-inter	D-E-CS	D-A-CS
T.Mult.	20,87	33,33	24,20	32,27	54,59
T.Mult.-noCS	20,93	32,67	23,54	31,03	53,61
T.Mult.-cur20	<b>20,55</b>	<b>31,92</b>	<b>20,01</b>	<b>26,70</b>	<b>53,06</b>

Due to restricted space we only display the multilingual results, however, the monolingual results are similar to the LSTM-based model, as well. In Table 9 it is possible to see that the general trends from the LSTM-based model also hold for the Transformer. The model performs worse than our LSTM architecture, however, this might be due to suboptimal hyperparameters as we focused on the LSTM model for our experiments.

## 7 Conclusion

In this work, we described a simple yet effective way of artificially generating CS data to improve on the inter-sentential CS task. We showed that our collected read-speech test data is more reliable for performance evaluation than using artificially generated test data. We also saw that the presented approach improves the monolingual performance of multilingual models, without any changes in the model architecture. More importantly, we enable a language-agnostic multilingual S2S model to automatically transcribe CS speech without providing any real CS data. Our experiments reveal that a model trained on artificial pseudo-CS data between language  $x \leftrightarrow y$  and  $y \leftrightarrow z$  is able to transcribe CS utterances with switches between languages  $x \leftrightarrow z$ . In such a scenario our model x-oenx (tenth row) Table 8 improves over the baseline multilingual model x-noCS by 5,03%WER, column D-A-CS. These results are especially important as there are millions of multilingual speakers code-switching in their everyday life. Thus systems able to process these inputs are much needed even when there is no data for all language pairs. In the future, we want to use language pairs from previous work in order to be able to compare to those as well.

## 8 Acknowledgement

The project on which this report is based was funded by the Federal Ministry of Education and Research (BMBF) of Germany under the numbers 01EF1803B (RE-LATER) and 01IS18040A (OML).

## References

1. Ahmed, B. & Tan, T. Automatic speech recognition of code switching speech using 1-best rescoring. *2012 International Conference On Asian Language Processing*. pp. 137-140 (2012)
2. Ali, A., Bell, P., Glass, J., Messaoui, Y., Mubarak, H., Renals, S. & Zhang, Y. The MGB-2 challenge: Arabic multi-dialect broadcast media recognition. *2016 IEEE Spoken Language Technology Workshop (SLT)*. pp. 279-284 (2016)
3. Amazouz, D., Adda-Decker, M. & Lamel, L. Addressing code-switching in French/Algerian Arabic speech. *Interspeech 2017*. pp. 62-66 (2017)
4. Ardila, R., Branson, M., Davis, K., Henretty, M., Kohler, M., Meyer, J., Morais, R., Saunders, L., Tyers, F. & Weber, G. Common voice: A massively-multilingual speech corpus. *ArXiv Preprint ArXiv:1912.06670*. (2019)
5. Bengio, Y., Louradour, J., Collobert, R. & Weston, J. Curriculum learning. *Proceedings Of The 26th Annual International Conference On Machine Learning*. pp. 41-48 (2009)
6. Chan, J., Ching, P. & Lee, T. Development of a Cantonese-English code-mixing speech corpus. *Ninth European Conference On Speech Communication And Technology*. (2005)
7. Chang, C., Chuang, S. & Lee, H. Code-switching sentence generation by generative adversarial networks and its application to data augmentation. *ArXiv Preprint ArXiv:1811.02356*. (2018)
8. Chorowski, J., Bahdanau, D., Serdyuk, D., Cho, K. & Bengio, Y. Attention-based models for speech recognition. *ArXiv Preprint ArXiv:1506.07503*. (2015)
9. Statistische Bundesamt (2022) Ausländische Bevölkerung nach Bundesländern und Jahren. DESTATIS.  
<https://www.destatis.de/DE/Themen/Gesellschaft-Umwelt/Bevoelkerung/Migration-Integration/Tabellen/auslaendische-bevoelkerung-bundeslaender-jahre.html#fussnote-1-116852> Cited 30 Jan 2023
10. Di Gangi, M., Cattoni, R., Bentivogli, L., Negri, M. & Turchi, M. Must-c: a multilingual speech translation corpus. *2019 Conference Of The North American Chapter Of The Association For Computational Linguistics: Human Language Technologies*. pp. 2012-2017 (2019)
11. Diwan, A., Vaideeswaran, R., Shah, S., Singh, A., Raghavan, S., Khare, S., Unni, V., Vyas, S., Rajpuria, A., Yarra, C. & Others Multilingual and code-switching ASR challenges for low resource Indian languages. *ArXiv Preprint ArXiv:2104.00235*. (2021)
12. Du, C., Li, H., Lu, Y., Wang, L. & Qian, Y. Data augmentation for end-to-end code-switching speech recognition. *2021 IEEE Spoken Language Technology Workshop (SLT)*. pp. 194-200 (2021)
13. Gage, P. A new algorithm for data compression. *C Users Journal*. **12**, 23-38 (1994)
14. Graves, A., Fernández, S., Gomez, F. & Schmidhuber, J. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. *Proceedings Of The 23rd International Conference On Machine Learning*. pp. 369-376 (2006)
15. Graves, A. Sequence transduction with recurrent neural networks. *ArXiv Preprint ArXiv:1211.3711*. (2012)
16. Guo, P., Xu, H., Xie, L. & Chng, E. Study of semi-supervised approaches to improving english-mandarin code-switching speech recognition. *ArXiv Preprint ArXiv:1806.06200*. (2018)
17. Hamed, I., Denisov, P., Li, C., Elmahdy, M., Abdennadher, S. & Vu, N. Investigations on speech recognition systems for low-resource dialectal Arabic-English code-switching speech. *Computer Speech Language*. **72** pp. 101278 (2022)
18. Hochreiter, S. & Schmidhuber, J. Long short-term memory. *Neural Computation*. **9**, 1735-1780 (1997)
19. Huber, C., Hussain, J., Nguyen, T., Song, K., Stüker, S. & Waibel, A. Supervised adaptation of sequence-to-sequence speech recognition systems using batch-weighting. *Proceedings Of The 2nd Workshop On Life-long Learning For Spoken Language Systems*. pp. 9-17 (2020)



20. Hussein, A., Chowdhury, S., Abdelali, A., Dehak, N. & Ali, A. Code-Switching Text Augmentation for Multilingual Speech Processing. *ArXiv Preprint ArXiv:2201.02550*. (2022)
21. Hussain, J., Mediani, M., Behr, M., Cheragui, M., Stüker, S. & Waibel, A. German-Arabic Speech-to-Speech Translation for Psychiatric Diagnosis. *Proceedings Of The Fifth Arabic Natural Language Processing Workshop*. pp. 1-11 (2020,12), <https://aclanthology.org/2020.wanlp-1.1>
22. Khosravani, A., Garner, P. & Lararidis, A. An Evaluation Benchmark for Automatic Speech Recognition of German-English Code-Switching. *2021 IEEE Automatic Speech Recognition And Understanding Workshop (ASRU)*. pp. 811-816 (2021)
23. Kingma, D. & Ba, J. Adam: A method for stochastic optimization. *ArXiv Preprint ArXiv:1412.6980*. (2014)
24. Koehn, P. & Others Europarl: A parallel corpus for statistical machine translation. *MT Summit*. 5 pp. 79-86 (2005)
25. LeCun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W. & Jackel, L. Handwritten digit recognition with a back-propagation network. *Advances In Neural Information Processing Systems*. 2 (1989)
26. Lee, D., Kim, D., Yun, S. & Kim, S. Phonetic Variation Modeling and a Language Model Adaptation for Korean English Code-Switching Speech Recognition. *Applied Sciences*. 11, 2866 (2021)
27. Li, K., Li, J., Ye, G., Zhao, R. & Gong, Y. Towards code-switching ASR for end-to-end CTC models. *ICASSP 2019-2019 IEEE International Conference On Acoustics, Speech And Signal Processing (ICASSP)*. pp. 6076-6080 (2019)
28. Luo, N., Jiang, D., Zhao, S., Gong, C., Zou, W. & Li, X. Towards end-to-end code-switching speech recognition. *ArXiv Preprint ArXiv:1810.13091*. (2018)
29. Lyu, D., Tan, T., Chng, E. & Li, H. An analysis of a Mandarin-English code-switching speech corpus: SEAME. *Age*. 21 pp. 25-8 (2010)
30. Mesthrie, R. *Introducing sociolinguistics*. (Edinburgh University Press,2009)
31. Müller, M., Stüker, S. & Waibel, A. Language adaptive multilingual CTC speech recognition. *International Conference On Speech And Computer*. pp. 473-482 (2017)
32. Mussakhojayeva, S., Khassanov, Y. & Atakan Varol, H. A Study of Multilingual End-to-End Speech Recognition for Kazakh, Russian, and English. *International Conference On Speech And Computer*. pp. 448-459 (2021)
33. Nakayama, S., Tjandra, A., Sakti, S. & Nakamura, S. Speech chain for semi-supervised learning of japanese-english code-switching asr and tts. *2018 IEEE Spoken Language Technology Workshop (SLT)*. pp. 182-189 (2018)
34. Nakayama, S., Tjandra, A., Sakti, S. & Nakamura, S. Zero-shot code-switching ASR and TTS with multilingual machine speech chain. *2019 IEEE Automatic Speech Recognition And Understanding Workshop (ASRU)*. pp. 964-971 (2019)
35. Nguyen, T., Stueker, S., Niehues, J. & Waibel, A. Improving sequence-to-sequence speech recognition training with on-the-fly data augmentation. *ICASSP 2020-2020 IEEE International Conference On Acoustics, Speech And Signal Processing (ICASSP)*. pp. 7689-7693 (2020)
36. Peddinti, V., Wang, Y., Povey, D. & Khudanpur, S. Low latency acoustic modeling using temporal convolution and LSTMs. *IEEE Signal Processing Letters*. 25, 373-377 (2017)
37. Poplack, S. Sometimes i'll start a sentence in spanish y termino en espanol: toward a typology of code-switching1. (Walter de Gruyter, Berlin/New York Berlin, New York,1980)
38. Pratapa, A., Bhat, G., Choudhury, M., Sitaram, S., Dandapat, S. & Bali, K. Language modeling for code-mixing: The role of linguistic theory based synthetic data. *Proceedings Of The 56th Annual Meeting Of The Association For Computational Linguistics (Volume 1: Long Papers)*. pp. 1543-1553 (2018)
39. Rousseau, A., Deléglise, P. & Esteve, Y. TED-LIUM: an Automatic Speech Recognition dedicated corpus.. *LREC*. pp. 125-129 (2012)
40. Sanabria, R., Caglayan, O., Palaskar, S., Elliott, D., Barrault, L., Specia, L. & Metze, F. How2: a large-scale dataset for multimodal language understanding. *ArXiv Preprint ArXiv:1811.00347*. (2018)

41. Schultz, T. & Waibel, A. Experiments on cross-language acoustic modeling.. *INTER-SPEECH*. pp. 2721-2724 (2001)
42. Schultz, T. & Waibel, A. Language-independent and language-adaptive acoustic modeling for speech recognition. *Speech Communication*. **35**, 31-51 (2001)
43. Seki, H., Watanabe, S., Hori, T., Le Roux, J. & Hershey, J. An end-to-end language-tracking speech recognizer for mixed-language speech. *2018 IEEE International Conference On Acoustics, Speech And Signal Processing (ICASSP)*. pp. 4919-4923 (2018)
44. Shah, S., Abraham, B., Sitaram, S., Joshi, V. & Others Learning to recognize code-switched speech without forgetting monolingual speech recognition. *ArXiv Preprint ArXiv:2006.00782*. (2020)
45. Shan, C., Weng, C., Wang, G., Su, D., Luo, M., Yu, D. & Xie, L. Investigating end-to-end speech recognition for mandarin-english code-switching. *ICASSP 2019-2019 IEEE International Conference On Acoustics, Speech And Signal Processing (ICASSP)*. pp. 6056-6060 (2019)
46. Stuker, S., Schultz, T., Metze, F. & Waibel, A. Multilingual articulatory features. *2003 IEEE International Conference On Acoustics, Speech, And Signal Processing, 2003. Proceedings (ICASSP'03)*. **1** pp. I-I (2003)
47. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A., Kaiser, Ł. & Polosukhin, I. Attention is all you need. *Advances In Neural Information Processing Systems*. **30** (2017)
48. Waibel, A., Hanazawa, T., Hinton, G., Shikano, K. & Lang, K. Phoneme recognition using time-delay neural networks. *IEEE Transactions On Acoustics, Speech, And Signal Processing*. **37**, 328-339 (1989)
49. Watanabe, S., Hori, T. & Hershey, J. Language independent end-to-end architecture for joint language identification and speech recognition. *2017 IEEE Automatic Speech Recognition And Understanding Workshop (ASRU)*. pp. 265-271 (2017)
50. Weiner, J., Vu, N., Telaar, D., Metze, F., Schultz, T., Lyu, D., Chng, E. & Li, H. Integration of language identification into a recognition system for spoken conversations containing code-switches. *Spoken Language Technologies For Under-Resourced Languages*. (2012)
51. Yilmaz, E., Heuvel, H. & Van Leeuwen, D. Investigating bilingual deep neural networks for automatic recognition of code-switching frisian speech. *Procedia Computer Science*. **81** pp. 159-166 (2016)
52. Yilmaz, E., Heuvel, H. & Leeuwen, D. Acoustic and textual data augmentation for improved ASR of code-switching speech. *ArXiv Preprint ArXiv:1807.10945*. (2018)
53. Zeng, Z., Khassanov, Y., Pham, V., Xu, H., Chng, E. & Li, H. On the end-to-end solution to mandarin-english code-switching speech recognition. *ArXiv Preprint ArXiv:1811.00241*. (2018)
54. Zhang, S., Yi, J., Tian, Z., Tao, J., Yeung, Y. & Deng, L. Reducing language context confusion for end-to-end code-switching automatic speech recognition. *ArXiv Preprint ArXiv:2201.12155*. (2022)