Blind Language Separation: Disentangling Multilingual Cocktail Party Voices by Language

Marvin Borsdorf 1, Kevin Scheck2, Haizhou Li3, Tanja Schultz2

1 Machine Listening Lab (MLL), University of Bremen, Germany
2 Cognitive Systems Lab (CSL), University of Bremen, Germany
3 The Chinese University of Hong Kong, Shenzhen, China

marvin.borsdorf@uni-bremen.de, scheck@uni-bremen.de

Abstract

We introduce blind language separation (BLS) as novel research task, in which we seek to disentangle overlapping voices of multiple languages by language. BLS is expected to separate seen as well as unseen languages, which is different from the target language extraction task that works for one seen target language at a time. To develop a BLS model, we simulate a multilingual cocktail party database, of which each scene consists of two randomly selected languages, each represented by two randomly selected speakers. The database follows the recently proposed GlobalPhoneMCP database design concept that uses the audio data of the GlobalPhone 2000 Speaker Package. We show that a BLS model is able to learn the language characteristics so as to disentangle overlapping voices by language. We achieve a mean SI-SDR improvement of 12.63 dB over 231 test sets. The performance on the individual test sets varies depending on the language combination. Finally, we show that BLS can generalize well to unseen languages and languages in the mixture.

Index Terms: Blind language separation, selective auditory attention, multilingual, GlobalPhone, cocktail party problem

1. Introduction

Selective auditory attention, also referred to as the cocktail party problem [1], describes the human ability to disentangle overlapping voices or sounds into the individual sources for arbitrary soundscapes. Speech separation represents the research field that attempts to equip machines with such an ability. The study of speech separation seeks to separate the individual voices in a speech mixture, that is called blind source separation [2–11] or target speaker extraction [12–20].

In blind source separation, a speech separation model disentangles all voices in a given overlapping speech mixture in one step. This usually requires the number of individual voices to be known in advance, because such systems are tied to a fixed number of voices or, more precisely, output channels. In real-world everyday conversational situations, this information is not always accessible and also may vary. In order to address this constraint, the research in this field proposed to estimate a maximum number of sources [5, 21, 22], to convert the reconstructed sources into a sequential output stream [23], or to apply recursive or iterative approaches [24, 25]. Blind source separation faces the permutation problem which is caused by the ambiguous assignment of sources to the output channels. The permutation invariant training (PIT) [4, 5] has been introduced as a method to solve this problem.

In target speaker extraction, a single speaker’s voice is isolated from the remaining sources of a given cocktail party mixture. To achieve this, the extraction model utilizes a reference signal of the desired speaker, that is typically given as additional speech signal. Therefore, target speaker extraction is not restricted by the outlined limitations.

The required databases to develop such models usually provide data tuples consisting of an input mixture of overlapped speech, the corresponding isolated clean sources, and, for target speaker extraction, an additional reference signal. While speech separation has been widely studied on monolingual databases in English language, e.g., on WSJ0-2Mix [2], WSJ0-3Mix [2], LibriCSS [26], WHAM! [27], WHAMR! [28], and LibriMix [29], recently multilingual studies have been performed, as in Apeltans et al. [30] and on GlobalPhoneMS2 [31].

Recently, target language extraction (TLE) [32] has been introduced as novel task that considers the cocktail party problem as multilingual scene. In TLE, a model isolates the voices of all speakers who talk in the target language from the remaining voices at once, independent of the interfering language and the number of speakers in the mixture. For this purpose, GlobalPhoneMCP [32] has been introduced as a new corpus design concept to simulate multilingual cocktail party mixtures of two and four speakers. While previous techniques on blind source separation and target speaker extraction, except for X-TaSNet [17], treated the output and ground truth data as single sources, TLE extracts mixture signals comprising one or two speakers. Since a TLE model is tied to a fixed target language by design, multiple TLE models are required to extract voices of different languages. To remove this limitation, multi-target language extraction (Multi-TLE) [33] models have been developed which can extract the voices for different target languages, based on an additional conditioning input that serves as language switch. However, both TLE and Multi-TLE models can only extract voices for a single target language at a time.

In this work, we introduce a novel task called blind language separation (BLS) that builds the bridge between blind source separation and TLE. BLS seeks to disentangle multilingual cocktail party voices into speaker groups by language in a single step. In comparison to TLE, a BLS model is not conditioned on the to-be-extracted target language, but performs the separation task using the languages’ characteristics that are given in the mixture. Consequently, a BLS model is also able to disentangle voices by language, even if the languages were unseen during training. Furthermore, a single BLS model can replace several applications of TLE or Multi-TLE on the same multilingual speech mixture in order to output different monolingual mixtures. This allows to reduce the execution time when compared to subsequent TLE applications or the memory footprint in comparison to a parallel TLE processing. We establish baselines for the novel BLS task by training seven different models that are based on the Conv-TasNet [8] and
SepFormer [11] architectures. We investigate different training language combinations and evaluate all models on 231 different test sets in total. For training the models, we adapt the recent dynamic language mixing algorithm [33] and create training data consisting of multilingual speech mixtures and language-specific speech mixtures on-the-fly.

The rest of the paper is organized as follows. In Section 2, we introduce the BLS task and provide details on the design. In Section 3, we describe the experimental setup, followed by Section 4, in which we show and discuss the results. In Section 5, we conclude our study and talk about our future work. We will make all information and scripts publicly available1.

2. Blind Language Separation

The recent results on TLE and Multi-TLE show that it is possible to equip neural network based models with the capability to extract voices of a specific target language from a multilingual cocktail party mixture [32, 33]. More precisely, the mechanism uses language characteristics to extract all target voices, independent of the interfering language and the number of speakers in the mixture, at once (see Figure 1 - bottom). This represents a novel approach to work on the solution of the cocktail party problem. However, TLE and Multi-TLE can only extract the voices for one target language at a time. In blind source separation (BSS), speaker characteristics are used to disentangle all overlapping speakers of a mixture into the individual sound sources in one step (see Figure 1 - top). With the introduction of blind language separation (BLS), we build the bridge between both aforementioned mechanisms to disentangle the overlapping voices of a multilingual cocktail party into the contributing speaker groups by language in a single step. The groups are given as monolingual voice mixtures (see Figure 1 - middle).

It has been shown that TLE and Multi-TLE models can be developed by applying a supervised training scheme with appropriate data tuples of a well constructed database. We extend this method with BSS-specific training mechanisms to realize a BLS development setup. In this setup, a BLS model processes a multilingual cocktail party input mixture and estimates a mask for each contributing, intra-language speaker group. Those masks are used to filter out the respective speaker groups’ voices which are given in the corresponding language. Similar to BSS, the model outputs the target signals on different output channels in parallel and thus requires the number of languages in the mixture to be known in advance. As the design concepts of BSS and, consequently, BLS face the permutation problem, we apply PIT that allows to find the best permutation of separated sources (for BSS) and mixture signals (for BLS). It is worth noting that if each language in the mixture would be represented by only one speaker, the BLS task would be equal to a BSS task as each output channel would comprise a single speaker’s voice. Since this would be contrary to our training concept, we constrain the BLS task to mixtures that contain at least one language that is represented by at least two speakers.

3. Experimental Setup

We perform experiments based on two different model architectures to study BLS. We evaluate the models under conditions that comprise unseen speakers as well as seen and unseen languages. Additionally, we investigate mixed conditions, i.e. the mixture comprises seen and unseen languages.

1https://github.com/mborsdorf/BlindLanguageSeparation

Figure 1: Comparison of blind language separation with target language extraction and blind source separation.

3.1. Data setup

We describe the data source, the language mixing strategy, and the data design for our experiments.

Corpus The data in our study is based on the multilingual GlobalPhone corpus that has been proposed by Schultz [34] and Schultz et al. [35]. More precisely, we utilize the speech data of a subset, called GlobalPhone 2000 Speaker Package2, which is distributed by the European Language Resources Association (ELRA) [36] under research and commercial licenses. The subset covers 22 languages with 2,000 native speakers, each one represented with roughly 40 seconds of speech. The included languages are: Arabic (AR), Bulgarian (BG), Chinese Mandarin (CH), Chinese Shanghai (WU), Croatian (HR), Czech (CZ), French (FR), German (DE), Hausa (HA), Japanese (JA), Korean (KO), Polish (PL), Portuguese (PT), Russian (RU), Spanish (SP), Swahili (SA), Swedish (SW), Tamil (TA), Thai (TH), Turkish (TU), Ukrainian (UA), and Vietnamese (VN). To split the data into training, validation, and test sets, and to create the multilingual mixtures, we apply the GlobalPhoneMCP corpus design concept [32]. The training and validation sets share the same speakers, but with different utterances. The test set speakers are disjoint from those sets to allow an evaluation under unseen speaker conditions. We reduce the computational costs by downsampling the audio data to 8 kHz.

Dynamic language mixing procedure To simulate the multilingual cocktail party audio mixtures, we adapt the dynamic language mixing procedure [33] as follows (see Figure 2). First, we sample L languages from a fixed set of languages. Second, for each language, we sample a random number of speakers. Third, for each speaker, we select a random utterance. Fourth, we sample signal-to-noise ratios (SNRs) with zero mean and, under consideration of the respective active speech levels, we create intra-language mixtures for each language. It is worth noting that we do not normalize the signals based on the maximum amplitude in this step. The intra-language mixtures represent the ground truth data for PIT. Fifth, we sample SNR values to simulate the overall mixture that is given as a weighted sum of all intra-language mixtures. The final overall mixture represents the multilingual cocktail party mixture, which is also used as the input to the model. The intra-language mixtures and the overall mixture are normalized based on the maximum amplitude. When creating the intra-language and overall mixtures, we cut the longer signal to the length of the shorter one, which is commonly referred to as min version. Before performing the adapted dynamic language

2https://catalog.elra.info/en-us/repository/browse/ELRA-S0400/
mixing procedure, we apply scripts that have been released by Isik et al. [37] to downsample the audio data and to determine and store the active speech level for each speaker’s utterances.

The adapted dynamic language mixing for BLS can be used to create mixtures with an arbitrary number of languages and speakers. However, as this work represents the first attempt at BLS, we study conditions with a fixed number of languages and speakers. We set the number of languages and the number of speakers per language to two, leading to a total of four speakers in the mixture. With these settings, we fulfill the requirement of having at least one language that comprises more than one speaker (see Section 2). During training and validation, we apply the adapted dynamic language mixing method to create data tuples of input mixture and ground truth data on-the-fly. The SNRs to simulate the intra-language and overall mixtures and, subsequently, the overall mixture.

**Test data** For creating the test data, we use a static version of the adapted dynamic language mixing routine, which guarantees comparability between different studies. To simulate the data, we perform the following steps. First, we create 360 intra-language speaker pairs for each language (following the test set size in [32]) and balance the repetitions of each utterance as far as possible. Second, we randomly mix the intra-language speaker pairs of two languages to create overall mixtures. We repeat this for all unique language combinations of the 22 languages in the GlobalPhone 2000 Speaker Package, without same-language pairs. This leads to 231 test sets with 83,160 utterances in total. We make sure to avoid duplicates in combinations of intra-language and overall mixtures. We apply the same steps to sample SNRs and normalize the data as for training and validation. Since each language is represented with a different number of utterances in the GlobalPhone 2000 Speaker Package, this creates an imbalance in the contribution of each individual utterance in the intra-language and overall mixtures. For example, single utterances of WU (16 speakers and 80 utterances) may influence the results more than utterances of HA (23 speakers and 179 utterances). However, we favor this approach to the alternative of using a fixed number of utterances per language, such as using only 80 utterances, because this would discard a substantial amount of test data.

**3.2. Network architectures**

In our study, we train BLS models based on the Conv-TasNet [8] and SepFormer [11] time-domain architectures. The architectures estimate a mask for each output channel, in which every channel corresponds to an intra-language mixture in the overall input mixture. For the Conv-TasNet model, we set the hyper-parameters according to the best-performing model in Luo and Mesgarani [8], except that we change the mask activation to rectified linear unit (ReLU). The implementation is done with help of the Asteroid [38] toolkit. In total, the model comprises 5.05 million trainable parameters. For the SepFormer model, we adopt the best-performing settings in Subakan et al. [11], resulting in 25.68 million trainable parameters, and use the SpeechBrain [39] toolkit implementation.

**3.3. Training and evaluation**

In our experiments, we use five training sets, each defined by a different composition of languages. The sets comprise two, three, five, ten, and eighteen randomly chosen languages, denoted as L-2, L-3, L-5, L-10, and L-18 respectively. The sets are subsets of each other, i.e., larger sets include the languages of smaller sets. The languages CH, CR, JA, and PL are only part of the test data, such that we can evaluate the models under seen, unseen, and mixed conditions. The speakers in the test data are always unseen.

We train five Conv-TasNet models, each on one training set, to establish first results. We further train two instances of SepFormer on ten and eighteen languages respectively, to investigate whether BLS models with a larger number of parameters can benefit from more languages in the training set. For training, we provide data tuples comprising overall input mixture and intra-language mixtures and apply a supervised training scheme using PIT. We use Adam [40] as optimizer with initial learning rates of $1\times10^{-3}$ and $1.5\times10^{-3}$ for Conv-TasNet and SepFormer respectively. The gradient clipping is set to five and the weight decay to zero for both architectures. We train the Conv-TasNet on batches with a size of forty while the SepFormer is trained with a batch size of two.

We use the negative scale-invariant signal-to-distortion ratio (SI-SDR) [41] as training objective and validation loss. The positive SI-SDR improvement (dB) is used to evaluate the model’s performance during testing. To determine the latter, we calculate the difference between the SI-SDR of the separated signals (using the permutation that yields the maximum mean value) and the SI-SDR of the input mixture, both with respect to the ground truth signals. In order to monitor and control the training of our models, we utilize (a) a learning rate scheduler that halves the current learning rate if the validation loss does not decrease within three epochs and (b) an early stopping algorithm which stops the training if there is no improvement in the validation loss within six subsequent epochs. Each model is trained from scratch on the respective training set and the maximum number of training epochs is 200.

During training and validation, we use the adapted dynamic language mixing method to continuously create 20,000 and 10,000 random data tuples per epoch. We split the data into chunks of four seconds. While samples with a duration between two and four seconds are padded with zeros, samples with a duration of less than two seconds are neglected. The evaluation is performed on the 231 pre-simulated test sets and the samples are processed on their entire length at once.

**4. Results and Discussion**

We report the blind language separation results for seven models in terms of average SI-SDR improvement (dB) over test samples of different language combinations. We illustrate the mean results over all 231 test sets (Mean,231), as well as some individual results. For the mixed conditions, we report the mean performances over GE-\{CH, CR, JA, PL\} and PO-\{CH, CR, JA, PL\} as GE mix and PO mix respectively (see Table 1).
Table 1: Blind language separation results for models trained on different language sets reported in terms of average SI-SDR improvement (dB) over all test samples of specific language combinations. With \textit{Mean}$_{231}$ we report the mean value over all 231 test sets. While \textit{seen} provides the results for the language combination GE-PO that is part of all training sets, \textit{unseen} shows the results for mixtures in which both languages have not been part of the training. \textit{Mixed (mean)} denotes the mean performance under mixed conditions, i.e. only one of the languages in the mixture is seen during training. More precisely, we report the mean performances over GE-\{CH, CR, JA, PL\} and PO-\{CH, CR, JA, PL\} as GE\_mix and PO\_mix respectively. The test set speakers are always unseen.

<table>
<thead>
<tr>
<th>Training set</th>
<th>BLS model</th>
<th>Mean$_{231}$ \textit{seen}</th>
<th>\textit{Mixed (mean)}</th>
<th>Unseen</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>GE-PO</td>
<td>Mixed _mix</td>
<td>PO_mix</td>
<td>CH-CR</td>
</tr>
<tr>
<td>L-2</td>
<td>Conv-TasNet</td>
<td>12.69</td>
<td>4.79</td>
<td>2.43</td>
</tr>
<tr>
<td>L-3</td>
<td>Conv-TasNet</td>
<td>12.00</td>
<td>4.30</td>
<td>4.10</td>
</tr>
<tr>
<td>L-5</td>
<td>Conv-TasNet</td>
<td>11.46</td>
<td>5.91</td>
<td>4.71</td>
</tr>
<tr>
<td>L-10</td>
<td>Conv-TasNet</td>
<td>6.85</td>
<td>11.28</td>
<td>6.54</td>
</tr>
<tr>
<td>L-10</td>
<td>SepFormer</td>
<td>11.00</td>
<td>15.16</td>
<td>8.64</td>
</tr>
<tr>
<td>L-18</td>
<td>Conv-TasNet</td>
<td>7.57</td>
<td>10.77</td>
<td>6.37</td>
</tr>
<tr>
<td>L-18</td>
<td>SepFormer</td>
<td>12.63</td>
<td>14.74</td>
<td>9.46</td>
</tr>
</tbody>
</table>

L-2: \{GE, PO\} \quad L-3: \{GE, PO\} \quad L-5: \{GE, PO\} \quad L-10: \{CH, CR, JA, PL\} \quad L-18: \{CH, CR, JA, PL\}

\textbf{Conv-TasNet} trained on two languages performs well under the seen languages condition. While the performance under the mixed condition GE\_mix is moderate, it is worse under PO\_mix. We assume that the model is slightly biased towards GE and can better estimate GE-specific masks for separation. Recent studies on Multi-TLE show similar findings in the comparison of GE and PO [33]. The individual results under the unseen conditions vary strongly, showing that the model only works for some of those language combinations. The negative or rather corrupting performances are similar to recent findings in Borsdorf et al. [31]. Listening to the reconstructed audio of the test sets with negative results reveals that occasionally three speakers are assigned to one output channel while the remaining speaker and artifacts/noise are assigned to the other output channel. We cannot find any language similarities w.r.t. phonetics (mono-phone level) that could help to interpret the results. We think that different recording conditions, even if the database was carefully recorded, may influence the performance.

Training on three languages shows slightly degraded results under the conditions which have shown good results when trained on two languages. Under almost all other conditions, the performance increases, except for JA-PL. The performances for GE\_mix and PO\_mix are now similar. Training on five languages further raises the Mean$_{231}$ performance and improves the performance under mixed and unseen (CR-JA, CR-PL and JA-PL) conditions. We attribute this to an increase in the number of training languages and, consequently, in the number of speakers, utterances, and recording conditions.

Training on ten and eighteen languages improves the performance under most conditions, especially the CH-CR condition. The largest overall drop is given under the GE-PO condition. A comparison between the L-18 Conv-TasNet model and the L-2 model, which could be seen as an expert for this condition, shows a degradation of about 1.92 dB. We clearly see, that an overall improvement (indicated by 7.57 dB for Mean$_{231}$) comes at the cost of some individual performances.

Training SepFormer models on L-10 and L-18 sets outperforms the respective Conv-TasNet models. This is in line with recent results for different TLE architectures [32]. With a Mean$_{231}$ result of 12.63 dB, the L-18 SepFormer model yields the best overall performance. However, for some individual language combinations, the L-10 SepFormer model outperforms the L-18 model. We also indicate that the gain in Mean$_{231}$ performance between the L-10 and L-18 SepFormer models is larger compared to their L-10 and L-18 Conv-TasNet counterparts. The stronger separation performance in general as well as the larger performance gain may be based on the significantly larger amount of model parameters (approx. 5 million vs. 26 million parameters). In addition, we assume that the context information processing of the SepFormer could help to better identify language-specific characteristics in the input mixture, which may contribute to the superior language-based separation performance of the model.

5. Conclusion and Future Work

In this paper, we have introduced the novel blind language separation (BLS) task, which disentangles multilingual cocktail party voices into speaker groups by language. We established baseline BLS models which can separate the voices based on the language without any language conditioning. Models trained on more languages show a better performance on average and generalize well to unseen speakers and languages. However, the models show low performances for some language combinations. The current BLS models assume a fixed number of languages in the mixture and thus cannot disentangle multilingual mixtures if the number of languages is not known in advance. To counteract this problem, we will investigate recent blind source separation methods that have been proposed to address this limitation [5, 21–25] for BLS. We will combine this objective with a larger-scale study on language separation by including more languages. We also intend to find methods to analyze the imbalance of the separation performance.

6. Acknowledgements

This research is funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany’s Excellence Strategy (University Allowance, EXC 2077, University of Bremen).
7. References


