

master's thesis

# Multilingual Speech Recognition for Selected West-European Languages

*Bc. Gloria María Montoya Gómez*



May 25, 2018

advisor: Doc. Ing. Petr Pollák, CSc.

Czech Technical University in Prague  
Faculty of Electrical Engineering, Department of Circuit Theory

## I. Personal and study details

Student's name: **Montoya Gómez Gloria María** Personal ID number: **464314**  
Faculty / Institute: **Faculty of Electrical Engineering**  
Department / Institute: **Department of Circuit Theory**  
Study program: **Communications, Multimedia, Electronics**  
Branch of study: **Systems of Communication**

## II. Master's thesis details

Master's thesis title in English:

**Multilingual Speech Recognition for Selected West-European Languages**

Master's thesis title in Czech:

**Multilingvální rozpoznávání řeči pro vybrané západoevropské jazyky**

Guidelines:

1. Meet the principles of speech recognition with a special focus on multilingual systems based on GMM-HMM architecture.
2. Define unified shared phonetic alphabet for selected West-European languages based on X-SAMPA representation that can then be used for multilingual acoustic modeling.
3. Using KALDI toolkit, create multilingual acoustic models for languages covered in available speech databases and based on KALDI conventions, create example scripts (recipes) for the task of multilingual speech recognition.
4. Analyze achieved recognition accuracy using the data from available speech databases.

Bibliography / sources:

- [1] X. Huang, A. Acero, H.-W. Hon: Spoken Language Processing. Prentice Hall, 2001.
- [2] T. Schulz: Multilingual Speech Processing. Academic Press, 2006.
- [3] D. Jurafsky, J. H. Martin: Speech and Language Processing. 2nd edition. Prentice Hall, 2009.
- [4] J. Fiala: DNN-HMM Based Multilingual Recognizer of Telephone Speech. Diploma thesis, CTU FEE, 2016.
- [5] D. Povey et al: The Kaldi Speech Recognition Toolkit. In Proc. of IEEE 2011 ASRU, Hawaii, US, 2011. Note. Project WEB-page <http://kaldi.sourceforge.net/>.

Name and workplace of master's thesis supervisor:

**doc. Ing. Petr Pollák, CSc., Department of Circuit Theory, FEL**


Name and workplace of second master's thesis supervisor or consultant:

Date of master's thesis assignment: **31.01.2018** Deadline for master's thesis submission: \_\_\_\_\_

Assignment valid until: **30.09.2019**

  
\_\_\_\_\_  
doc. Ing. Petr Pollák, CSc.  
Supervisor's signature

  
\_\_\_\_\_  
prof. Ing. Pavel Sovka, CSc.  
Head of department's signature

  
\_\_\_\_\_  
prof. Ing. Pavel Ripka, CSc.  
Dean's signature

## III. Assignment receipt

The student acknowledges that the master's thesis is an individual work. The student must produce her thesis without the assistance of others, with the exception of provided consultations. Within the master's thesis, the author must state the names of consultants and include a list of references.

15/03/18  
Date of assignment receipt

Gloria María Montoya Gómez  
Student's signature

## **Acknowledgement**

I would like to thank my advisor, Doc. Ing. Petr Pollák, CSc. I was very privileged to have him as my mentor. Pollák was always friendly and patient to give me his knowledgeable advise and encouragement during my study at Czech Technical University in Prague. His high standards on quality taught me how to do good scientific work, write, and present ideas.

## **Declaration**

I declare that I worked out the presented thesis independently and I quoted all used sources of information in accord with Methodical instructions about ethical principles for writing academic thesis.

## Abstract

Hlavním cílem předložené práce bylo vytvoření první verze multilingválního rozpoznávače řeči pro vybrané 4 západoevropské jazyky. Klíčovým úkolem této práce bylo definovat vztahy mezi subslovními akustickými elementy napříč jednotlivými jazyky při tvorbě automatického rozpoznávače řeči pro více jazyků. Vytvořený multilingvální systém pokrývá aktuálně následující jazyky: angličtinu, němčinu, portugalsštinu a španělštinu. Jelikož dostupná fonetická reprezentace hlásek pro jednotlivé jazyky byla různá podle použitých zdrojových dat, prvním krokem této práce bylo její sjednocení a vytvoření sdílené fonetické reprezentace na bázi abecedy X-SAMPA. Pokud jsou dále acoustické subslovní elementy reprezentovány sdílenými skrytými Markovovy modely, případný nedostatek zdrojových dat pro trénování může být pokryt z jiných jazyků. Dalším krokem byla vlastní realizace multilingválního systému pomocí nástrojové sady KALDI. Použité jazykové modely byly na bázi zredukovaných trigramových modelů získaných z veřejně dostupných zdrojů. První experimenty byly realizovány pro monolingvální systémy pro výše zmíněné jazyky za účelem získání referenční informace o dosažitelné přesnosti. Následné použití sdíleného jazykového modelu napříč jazyky vedlo k určitému snížení přesnosti rozpoznávání, avšak tato byla nadále velmi vysoká. Nejmenší chyba na úrovni slov (WER) se pohybovala mezi 8.55% a 12.42% pro angličtinu a španělštinu. Další dosahované výsledky pro zbývající jazyky odpovídaly velikosti a kvalitě dostupných zdrojů pro získání akustických a jazykových modelů v navrženém rozpoznávacím systému.

## Klíčová slova

multilingvální rozpoznávání řeči, akustické modelování, GMM-HMM, rozpoznávání spojitě řeči s velkým slovníkem, LVCSR, KALDI, GlobalPhone, Wall Street Journal, X-SAMPA, IPA

## **Abstract**

The main goal of this work was to create the first version of a multilingual speech recognition system for selected four West-European languages. A crucial task of this work was to establish a relationship between subword acoustic units across particular languages which is the core for building of automatic speech recognition (ASR) system for multiple languages. The built multilingual ASR system, up to date, covers the following languages: English, German, Portuguese, and Spanish. Because the phonetic unit representation differed for particular language depending on the database used, the first step was intended to define a general shared phonetic representation based on X-SAMPA. When acoustic phonetic units represented by hidden Markov models (HMMs) are shared, the lack of certain missing training acoustic resources can be then complemented among languages. The following step was to implement the multilingual speech recognition system using KALDI speech recognition toolkit. Language models finally implemented were statistical pruned trigram ones and they were obtained from publicly available resources. The first experiments were carried out across monolingual systems to identify what recognition accuracy could be obtained. Further incorporation of the shared acoustic modeling yielded a reduction in term of accuracy, however, high accuracy results were still obtained. The best word error rates (WER) fluctuate between 8.55% and 12.42%. These values correspond to English and Spanish language respectively. Among the results, it was also found that for particular languages, the accuracy strongly depends on the size and quality of available resources for obtaining both acoustic and language models used in designed ASR system.

## **Keywords**

multilingual speech recognition, acoustic modeling, GMM-HMM, large vocabulary continuous speech recognition, LVCSR, KALDI, GlobalPhone, Wall Street Journal, X-SAMPA, IPA

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>HMM Based ASR framework</b>	<b>3</b>
2.1	Front-end: feature extraction . . . . .	3
2.1.1	Spectral shaping . . . . .	4
2.1.2	Spectral analysis . . . . .	5
2.1.3	Parametric transformations . . . . .	6
2.2	Back-end . . . . .	7
2.2.1	Acoustic model . . . . .	7
	HMM-Based Acoustic model . . . . .	7
	Unit and sequence modeling . . . . .	8
	Acoustic model transformation: Speaker adaptive training . . . . .	9
2.2.2	Lexicon . . . . .	9
2.2.3	Language model . . . . .	10
2.2.4	Decoder . . . . .	11
2.2.5	Speech recognition using WFST . . . . .	11
	Transducer composition . . . . .	11
	Training and decoding . . . . .	12
<b>3</b>	<b>ASR for multiple languages</b>	<b>13</b>
3.1	Multilingual ASR systems . . . . .	13
3.1.1	Lack of resources . . . . .	14
3.1.2	Acoustic model combination . . . . .	14
3.2	Language resources: Corpora and lexica . . . . .	14
3.2.1	TIMIT corpus . . . . .	15
3.2.2	Wall Street Journal corpus . . . . .	15
3.2.3	GlobalPhone database . . . . .	15
3.3	Phonetic alphabets . . . . .	16
3.3.1	The international phonetic alphabet . . . . .	16
3.3.2	Alphabet for English . . . . .	16
	ARPABET . . . . .	16
	TIMITBET . . . . .	18
	CMUBET . . . . .	19
3.3.3	Alphabet for German . . . . .	19
3.3.4	Alphabet for Portuguese . . . . .	20
3.3.5	Alphabet for Spanish . . . . .	20
3.3.6	Sharing phones . . . . .	20
3.4	Language models . . . . .	21
<b>4</b>	<b>Experimental part</b>	<b>26</b>
4.1	Language resources . . . . .	26
4.2	Evaluation metric . . . . .	27
4.3	Implementation of ASR in KALDI . . . . .	28
4.3.1	KALDI framework . . . . .	28
4.3.2	Setup of recipes stages . . . . .	28
	Stage 0: Data & lexicon & language preparation . . . . .	28
	Stage 1: MFCC Feature Extration & CMVN . . . . .	31

Stage 2: Acoustic model training . . . . .	31
Stage 3: Making graphs . . . . .	32
Stage 4: Decoding . . . . .	32
Arrangement of steps to run the multilingual system . . . . .	33
4.4 Results and discussions . . . . .	33
4.4.1 Preliminary test . . . . .	34
4.4.2 Monolingual ASR systems . . . . .	34
4.4.3 Shared acoustic modeling . . . . .	37
<b>5 Conclusions</b>	<b>40</b>
<b>Bibliography</b>	<b>42</b>
<b>Bibliography</b>	<b>42</b>

## Abbreviations

AM	Acoustic Model
ARPA	Advanced Research Projects Agency
ASR	Automatic Speech Recognition
CI	Context-Independent
CD	Context-Dependent
CMVN	Cepstral Mean and Variance Normalization
CSR	Continuous Speech Recognition
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
fMLLR	Feature Space Maximum Likelihood Linear Regression
HMM	Hidden Markov Model
IPA	International Phonetic Alphabet
LDA	Linear Discriminant Analysis
LVCSR	Large Vocabulary Continuous Speech Recognition
LDC	Linguistic Data Consortium
LM	Language Model
MAP	Maximum a Posteriori
MFCC	Mel Frequency Cepstral Coefficients
MLLT	Maximum likelihood linear transformation
OOV	Out-of-vocabulary
PDF	Probability Density Function
SAMPA	Speech Assessment Methods Phonetic Alphabet
SAT	Speaker Adaptive Training
SFSA	Stochastic Finite State Automata
SRI	Stanford Research Institute
SRILM	SRI Language Modeling
STFT	Short Time Fourier Transform
TI	Texas Instruments
WER	Word Error Rate
WFST	Weighted Finite State Transducers
WSJ	Wall Street Journal
X-SAMPA	Extended SAMPA



# 1 Introduction

Back in time, applications such as voice dialing, voice control, data entry or dictation included ASR applications. Nowadays, ASR technology has rapidly advanced because of the exponential growth of big data and computing power, and more challenging applications are becoming a reality. Instances are machine translation, in-vehicle navigation, etc. From these examples it can be discerned that more advanced speaker independent ASR applications capable of supporting several languages simultaneously are needed.

Despite ASR systems have almost reached human performance, there are many reasons why 100% accuracy in the recognition task cannot be achieved. Some challenges that have hampered such error-free recognition achievement are bound to the nature of speech. The absence of pauses between spoken words, local rates of speech within and across speakers in different contexts are some difficult tasks the recognizer has to deal with. In addition to this, the flexibility of the language and its size can also make the recognition task even more convoluted. For example, in large vocabulary continuous speech recognition (LVCSR) it is more likely to have more words that sound like each other and thus the distinction between words becomes more difficult to achieve.

Further difficulties can be found when a recognition system is intended to perform the recognition task for multiple languages, i.e, when the system is required not to be monolingual any longer. Challenges and labours such as the collection of large data resources (texts, voice recordings, pronunciation lexicons, and parsing grammar), memory constraints and response times are inevitably encountered. Gathering this information not only takes considerable time, but also financial resources since different fields of expertise are needed. It is said that building a LVCSR system requires dozens of hours of recording in order to ensure a dictionary coverage on the order of 100000 words, and matching pronunciation dictionaries to guide the decoding procedure. To efficiently deal with the enormous task of covering different languages while reducing the resource requirements has lastly been subject of study by the research community. Model-level tying techniques, usage of graphemes as lexical units, and pronunciation estimators, are some instances of the proposed solutions. The aim of this work is to explore an efficient approach, based on a unified global phone representation to eliminate the resource preliminary conditions by applying acoustic model combinations of those shared defined phones. Encouraging results can be found through a literature review in monolingual systems. Word error rates that just a few years ago were 14% have recently dropped to 5% [1]. Despite the unavoidable accuracy degradation when implementing shared parameters in multilingual systems [2], to achieve a WER as low as possible is the motivation of this recogniser design.

In the following section, chapter 2, it is presented the most prominent forming components of ASR systems as well as the theoretical foundations of the statistical speech recognition. Presentation of the original phonetic alphabet incorporated per language can be found in chapter 3, as well as the procedure to resolved the X-SAMPA conver-

## *1 Introduction*

sion on the basis of the International Phonetic Association (IPA). Chapter 4 provides a description of data used, the modifications and additions incorporated in the standard KALDI recipe, and the experiment results. The latest mentioned matter, experiment results, is subdivided according to the followed methodology to develop the system. The first subsection shortly shows the results obtained by a preliminary or introductory test. The second subsection presents the results that can be obtained when each database is independently used to build different monolingual systems, while the third subsection reflects in terms of accuracy the impact of implementing an acoustic shared model. To wrap this work thesis up, conclusions are presented in chapter 5 together with a short discussion of possible future work.

## 2 HMM Based ASR framework

In a statistical ASR approach, the goal is to find the most likely word sequence  $\widehat{W}$  given the acoustic observation  $X = \{x_1, \dots, x_t, \dots, x_T\}$ , where  $t$  denotes the frame number and  $T$  the total number of frames. This is formally state in Eq. (1) where  $\mathcal{W}$  is the set of all possible word sequences of a given vocabulary

$$\widehat{W} = \arg \max_{W \in \mathcal{W}} P(W|X) \quad (1)$$

From Eq. (1) it can be seen that speech recognition is formulated as a maximum a posteriori (MAP) decoding problem. Unfortunately, a posteriori probabilities cannot be directly calculated, consequently this probability has to be decomposed using Bayes rule. The restatement is shown in Eq. (2) where  $P(X|W)$  is likelihood of the acoustic observation sequence  $X$  given a word sequence  $W$ , and  $P(W)$  is the prior probability of a word sequence  $W$ .

$$\widehat{W} = \arg \max_{W \in \mathcal{W}} \underbrace{P(X|W)}_{\text{acoustic model}} \underbrace{P(W)}_{\text{language model}} \quad (2)$$

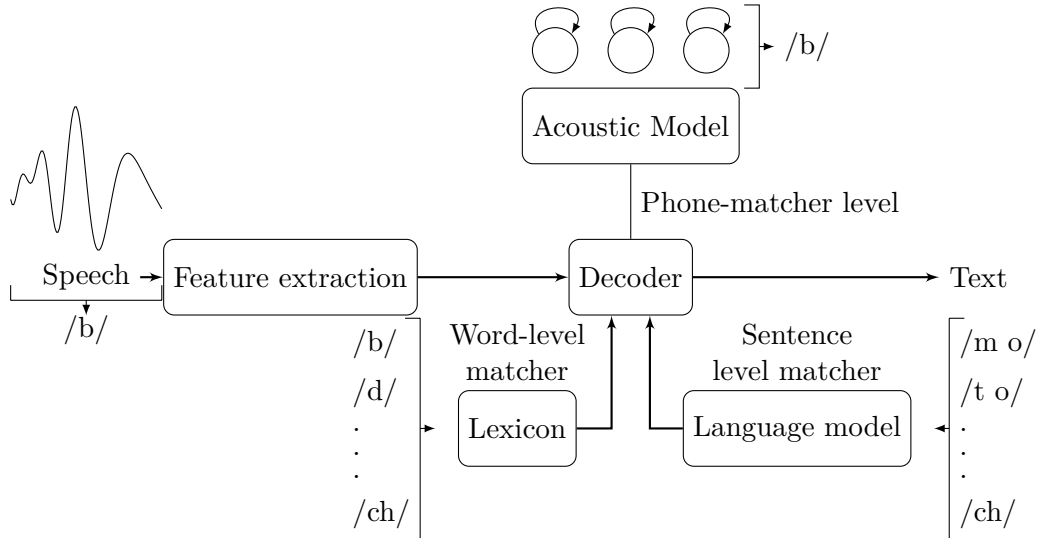
Technically speaking, the term  $P(X)$  should appear in the denominator; however, it is disregarded because it remains constant during the recognition [3]. The Eq. (2), also known as the fundamental equation of statistical speech recognition, defines the crucial constituents of this statistical speech recognition approach. The term on the left hand side is referred as the acoustic likelihood. The second term on the right hand side is the prior probability known as language model.

The architecture of a typical large vocabulary, speaker-independent, continuous speech recognition system is shown schematically in Fig. 1. It consists of a front-end module and a back-end module. In the next subsections the different processing stages of an ASR system are concisely described.

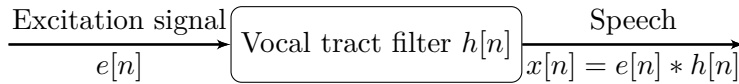
### 2.1 Front-end: feature extraction

Speech sound may be considered as a convolution of two independent components as shown in Fig. 2, where  $e[n]$  denotes the airflow at the vocal chords (excitation), and  $h[n]$  is the resonance of the vocal tract. The phonetic content is mostly dependent on the characteristics of the vocal tract filter. Thus the front-end module tasks are:

- Separation of the individual components. Technically, it alludes to the deconvolution of the source and the filter.
- Emphasize relevant properties of the acoustic signal for speech sound classification, while reducing redundant information.



**Figure 1** Basic architecture of a speech recognition system



**Figure 2** Source-Filter model of the speech signal.

Different techniques have been implemented in ASR systems to perform the aforementioned tasks. The processing steps of MFCC (Mel Frequency Cepstral Coefficient) feature estimation are illustrated in Fig. 3. A brief description of each of them is provided throughout the following sections.

### 2.1.1 Spectral shaping

#### Pre-emphasis

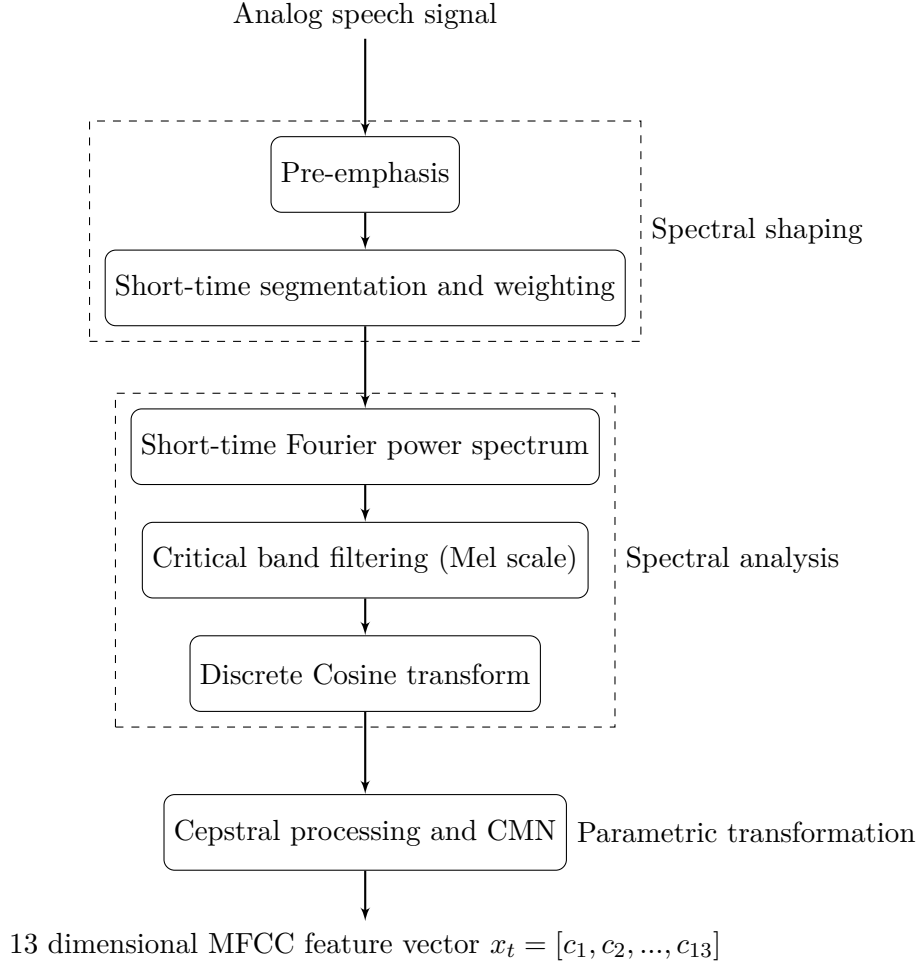
The spectrum of voiced sounds is characterized of a downward trend in which the frequencies in the upper part of the spectrum are attenuated due to the combination of the glottal source spectrum, and the radiation effect generated by the lips [4]. The pre-emphasis filter is intended to boost the high frequency of the signal spectrum approximately 20 dB/decade so that the negative spectral envelope is offset.

#### Short-time segmentation

The speech signal contains short portions of stationary characteristics within individual sounds. This means, over a short period of time, the statistics of the speech signal do not differ significantly from sample to sample. Such characteristic gives room to short-time based analysis [5]. The pre-emphasized signal is then sliced up into short-time segments, referred as frames. A length of 25 ms is typically used in the speech recognition field. The frame shift is usually 10 ms. A phone is assumed to last at least 30 ms, which add up to three frames [6].

#### Weighting

To avoid abrupt boundary discontinuities and spurious high-frequency components into



**Figure 3** MFCC feature extraction technique, which generates a 13-dimensional feature vector  $x_t$  for each frame.

the spectrum, each frame is then fed to a Hamming window. Today, in speech recognition, the Hamming window is almost exclusively used [7].

### 2.1.2 Spectral analysis

#### Short-time Fourier power spectrum

The standard discrete Fourier transform (DFT) is calculated for each weighted frame; technique that is named short-time discrete Fourier transform (STFT). MFCC feature extraction method relies on the spectrogram, which is the magnitude of the complex values calculated by the STFT. The complex spectral values  $X_m[k]$  of a weighted frame and the power spectrum  $G_m[k]$  are calculated using Eq. (3), where  $m$  represents the starting point for the localized DFT,  $k$  the DFT index of the segment, and  $N$  the length of the analysis window.

$$\begin{aligned}
 X_m[k] &= \sum_{n=0}^{N-1} x[n]w[n-m]e^{-j\frac{2\pi}{N}nk} \\
 G_m[k] &= \frac{|X_m[k]|^2}{N} = \frac{\text{Re}^2(X_m[k]) + \text{Im}^2(X_m[k])}{N}
 \end{aligned} \tag{3}$$

### Critical band filtering (Mel scale)

The Mel-filter bank is constructed by perceptual considerations: The human ear distinguishes lower frequencies at a much finer scale than higher frequencies [8]. A mel critical-band-like spectrum is obtained by integrating the multiplication between the power spectrum obtained from Eq. (3) and the Mel overlapping triangular weighting filters [9]. A dimensionality reduction is reached after this processing module. The set of coefficients that represent the spectral magnitude is typically around 30 [6].

### Log spectrum computation and inverse DFT

MFCCs are obtained by taking the inverse transform of the log Mel-scale filter bank parameters. This transformation can be performed by the Discrete Cosine Transform (DCT). The resulting vector from the DCT is truncated to retain only the low-order coefficients so that an accurate representation of the slowly varying vocal tract is kept [10]. Thus, it can be said that DCT calculates the deconvolution of the system presented in Fig. 2. The resulting representation is generally composed by 13 coefficients per frame ( $c_0, c_1, \dots, c_{12}$ ). The zeroth coefficient  $c_0$  is the sum of the log energies from each filter bank channel, considered also as a geometric measure of frame energy.

## 2.1.3 Parametric transformations

### Cepstral Mean Variance Normalization (CMVN)

Reducing the bias caused by time-invariant linear filtering is crucial to avoid a reduction in recognition task results [10]. CMVN is a feature-based noise compensation algorithm that comprises the cepstral mean subtraction (CMS) and cepstral variance normalization (CVN) methods. In a nutshell, CMS forces the average values of the cepstral coefficients per-utterance to be zero, and CVN reduces the mismatches by normalizing the second moment of the distribution of speech to a fixed value [11]. Both CMS and CVN can reduce energy dispersion caused by loudness across speakers, but cannot compensate energy variations within a single utterance. [12].

### Dynamic features

1. **Delta and acceleration:** The cepstral coefficients described so far are referred as static features since the dynamics of the spectral changes are not captured [5]. Temporal dynamics features are employed to introduce context at the feature extraction level through derivatives [13]. The first order derivative, called delta features ( $\Delta$ ), corresponds to the slope (or velocity). The second derivative, called delta-delta features ( $\Delta\Delta$ ), provides information about the curvature (or acceleration). First and second-order dynamic features are usually appended to the observation feature vector. The final vector, depending on what dynamic features are added, could take one of the following structures
  - 13 MFCC features (original)
  - 26 MFCC features (original + first derivative)
  - 39 MFCC features (original along with first and second derivative)
2. **Linear transformations LDA, MLLT:** Another approach to incorporate dynamics is to concatenate 9 to 13 neighboring feature vectors. This operation results in a high dimensional vector, increasing the complexity of the system. To decrease

the level of sophistication it is usual to perform a dimensionality reduction for all data followed by a maximization procedure. Linear discriminant analysis (LDA) and maximum likelihood linear transformation (MLLT) perform these two tasks, respectively. In other words, the consecutive feature frames are spliced to 40 dimensions using LDA and then the orthogonal transformation MLLT is applied to make the features more accurately modeled by diagonal covariance Gaussians [14].

## 2.2 Back-end

The back-end module performs the recognition task based on the input feature vectors. It relies on the information provided from three knowledge sources that are the acoustic model, the lexicon, and the language model.

### 2.2.1 Acoustic model

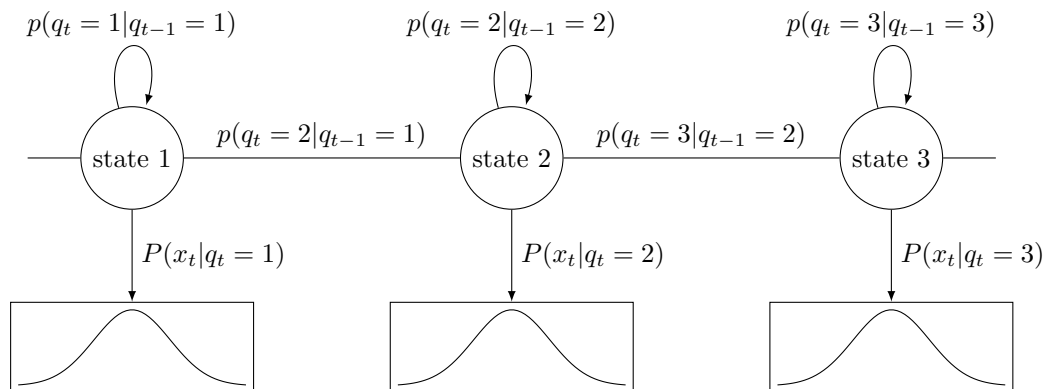
The acoustic model  $P(X|W)$  represents the sound units of a language based on speech features  $X$  extracted by the front-end processing [2]. The acoustic model is usually estimated by a Hidden Markov Model (HMM) [15], a kind of graphical model that represents the joint probability of an observed and a hidden (or latent) variable. The HMM is probably the most powerful statistical method for modeling speech signals [16]. Since this type of modeling is the cornerstone of the multilingual recognition system built in this thesis, an introduction of this framework is succinctly presented below.

#### HMM-Based Acoustic model

The HMMs used to represent the acoustic speech model are left-to-right in accordance with the Bakis model [9]. The state index of such model increases or remains unchanged as the time increases, leading to a move from left to right on the Markov chain. This translates to the model the causality of the speech production process. To mathematically put the HMM framework in the recognition task context, the fundamental question of statistical speech recognition (stated in Eq. (1)) can be analyzed as shown in Eq. (4). The most likely word sequence  $\widehat{W}$  presented before turns, in terms of this working structure, to the most likely state sequence  $Q^*$ .

$$\begin{aligned} Q^* &= \arg \max_{Q \in \mathcal{Q}} P(Q, X) \\ &= \arg \max_{Q \in \mathcal{Q}} \prod_{t=1}^T \underbrace{P(q_t|q_{t-1})}_{\text{transition score}} \underbrace{p(x_t|q_t)}_{\text{local emission score}} \end{aligned} \quad (4)$$

where  $\mathcal{Q}$  represents the set of all possible  $Q = \{q_1, \dots, q_t, \dots, q_T\}$  HMM state sequences. The right and left terms from Eq. (4) are technically referred as transition score or transition probability, and local emission score or emission probability respectively. Fig. 4 shows an example of such HMM model with its own transition emission probabilities. It also depicts the aforementioned concept of causality. The assumptions that have to be made when using a HMM statistical framework are:



**Figure 4** Example of three state and left-to-right HMM and its emission probability.

- Successive observations are assumed to be conditionally independent of past observations and states. This means that the probability that a particular acoustic vector emitted at time  $t$  depends only on the transition taken at that time, and it is conditionally independent of the past.
- The state chain is assumed to be first-order Markov. This means that the probability of being in a given state at time  $t$  only depends on the state at time  $t-1$  [17].

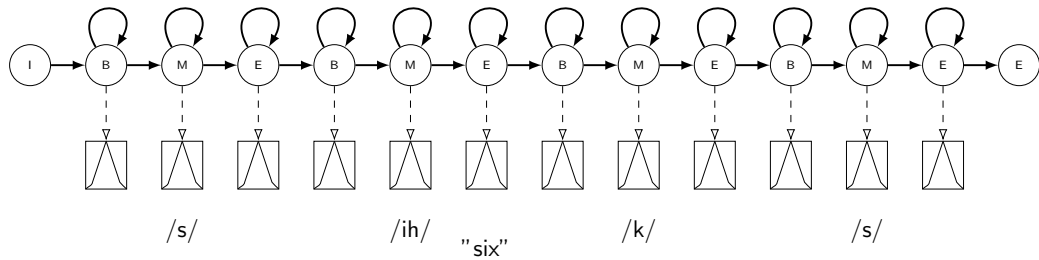
Gaussian mixture model (GMM) is popularly used to model the emission probability  $p(x_t|q_t)$  in a HMM-based automatic speech recognition system [18]. The Gaussian distribution is completely determined by two parameters, the mean  $\mu$  and the covariance matrix  $\Sigma$ . GMM approach, which is derived from a mixture of a finite Gaussian distributions, is preferred to a single Gaussian distribution since it allows a irregular distribution modeling.

### Unit and sequence modeling

HMM can represent different speech elements that can hierarchically be structured into phonemes, words, and sentences. A well structure model is primary since allows the system to potentially recognize an unlimited set of words, and directly impacts the extent of data sharing between acoustic models. Standardly, unit models represent phones or subphones (i.e. the beginning (B), middle (M), or end (E) part of phones) [6]. This is shown in Fig. 5. The implementation of this model adds acoustic context to the acoustic model. Depending on the context two subword units can be distinguished as follows

- **Context-independent subword unit** The acoustic unit set is defined based on the pronunciation lexicon. The number of acoustic units is  $D = K \times M$ , where  $K$  is the number of context-independent (CI) subword units in the lexicon and  $M$  is the number of HMM states for each context-independent subword unit, typically  $M = 3$  as shown in Fig. 5. In CI subword unit based ASR systems, the deterministic relationship between lexical and acoustic units is knowledge driven. Therefore, lexical model training is not involved, and the deterministic map between lexical and acoustic units is the lexical model [19].





**Figure 5** Representation of word "six" when the representation of beginning (B), middle (M), or end (E) is phone per phone

- Context-dependent sub-word unit based ASR systems** The physical articulators that produce sound cannot make rapid or large movements, thus they begin to move towards their target positions for the next phone while producing the current phone. Neighbouring articulator trajectories therefore overlap, affecting the acoustic realization of the current phone. In terms of ASR unit modeling this effect, called co-articulation, is considered using acoustic context-dependent (CD) subword units [8]. The number of CD acoustic units is  $I = M.K^{c_r+c_l+1}$  where  $c_l$  is the preceding context length, and  $c_r$  is the following context length. A typical CI model used is the triphone unit. This model has a distinct HMM for every unique pair of left and right.

Context dependent models are simple to build. The basic method starts with a set of single mixture monophone HMMs. These are cloned to form triphones and Baum-Welch re-estimation is used to train the triphone set. The number of mixture components in the triphones is gradually increased as in the monophone build case [8].

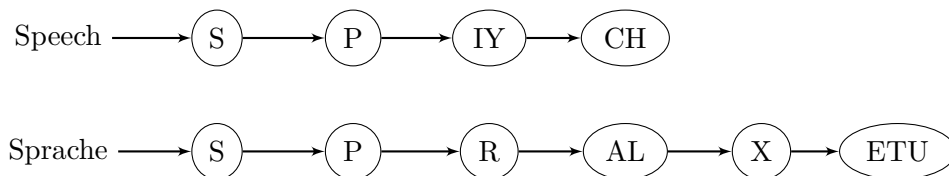
### Acoustic model transformation: Speaker adaptive training

The Speaker adaptive training (SAT) technique is used to reduce variation due to speaker, channel, or acoustic conditions. In few words, when SAT is applied the acoustic models are trained on speaker-normalized features. Then the inverse of the feature space maximum likelihood linear regression (fMLLR) matrix is used to remove the speaker identity from the original features [20]. The fMLLR transformation is an affine transformation of the features in the final 60-dimension recognition feature space that maximizes the likelihood of a speaker's data under an acoustic model [21]. In contrast to CMN where each feature vector component is processed separately, fMLLR makes use of a full transformation matrix that can be applied to the feature vector as a whole.

### 2.2.2 Lexicon

A lexicon or dictionary expresses the pronunciation of words in function of phones, the basic sounds in a language. Its role in the recognition task can be seen as a map from phoneme sequences to words. For optimal performance, the lexicon should list all

allowed or expected pronunciations of a word to be recognized. An example of a lexicon entry is presented in Fig. 6.



**Figure 6** A lexicon example. Phones are indicated by the symbols inside the ellipses.

Stochastic finite state automata (SFSA) are widely used to represent lexical and language models. In SFSA the set of legal word sequences is represented as a finite state network whose edges stand for the spoken words, i.e., each path through the network results in a legal word sequence [22]. The constituents of such representation are covered in Sec. 2.2.5.

### Unknown words

Out-of-vocabulary (OOV) words are unknown words that appear in the recognition task but not in the training vocabulary. A closed-vocabulary-based system can recognize only the words defined by the lexicon, and there will be no unknown words. An open dictionary, on the other hand, means that the system is able to model the unknown words by adding a pseudo-word usually called <UNK>.

### 2.2.3 Language model

The language model provides the apriori probability  $P(W)$  of a hypothesized word sequence independently of the acoustics. To include this information helps the speech recognizer to find the most likely word sequence when different sequences have the same acoustic likelihood  $P(X|W)$ , and reduces the search options during the decoding stage. A language model is specific to the corresponding language and independent from the character morphology modeled by HMMs [23]. Let the word sequence be  $W = (w_1, w_2, \dots, w_R)$ , the prior probability  $P(W)$  is given by

$$\begin{aligned}
 P(W) &= P(w_1, w_2, \dots, w_R) \\
 &= P(w_R | w_1, w_2, \dots, w_{R-1}) \\
 &= \prod_{k=1}^R P(w_k | w_1, \dots, w_{k-1})
 \end{aligned} \tag{5}$$

### Unigram, bigram and trigram

The N-gram language model estimates the probability  $P(W)$  by truncating the preceding word to  $N - 1$ . It can be estimated using the counts of the words from large training corpora. The maximum likelihood estimation is given then as follows

$$P(w_1^k) \approx \prod_{k=1}^R P(w_k | w_{k-N+1}, w_{k-N+2}, \dots, w_{k-1}) \tag{6}$$

Bigram alludes to pairs of adjacent tokens in a corpus. It can be defined in terms of any kind of linguistic unit, but are usually taken to be words. Bigram can be estimated when  $N = 2$  in Eq. 6. Unlike bigram, unigram (when  $N = 1$ ) and trigram (when  $N = 3$ ) are defined as a single word and a sequence of three words, respectively. The number of subsequent tokens is determined by the n-gram used.

### Back-off method

The Back-off method is used to prevent the language model to assign a probability 0 to unseen n-grams in the recognition phase that were not seen during the training. In short, this smoothing method artificially increases the number of all n-grams by 1, whether they occurred in the training corpus or not. By this all probabilities turn to be greater than 0 [24].

### 2.2.4 Decoder

The general assumption of the speech decoder is that the message carried in the speech signal is encoded as a sequence of symbols. Therefore, the task of the statistical decoder is to map the sequence of incoming feature vectors from the front-end module (Sec. 2.1) to the corresponding sequence of symbols. Mathematically speaking, the decoder target is to calculate the  $\arg \max_{W \in \mathcal{W}}$ . The statistical decoder has to cope with two problems that stem from the fixed rate that is used in the feature extraction stage. These are:

- Several speech signals with different lengths may carry the same message. As a consequence, there is no one-to-one mapping between the feature vectors and the sequence of symbols.
- A variety of feature vectors could yield to the same symbol. Given that the feature vectors are considered as samples of a stochastic process, the statistical decoder has to be able to characterize the common patterns of all feature vectors corresponding to a particular symbol.

A standard strategy applied in the recognition process is based on a beam Viterbi search. Initially, all possible words are added to the search beam. At each possible word boundary, the language model predicts the most likely subsequent word(s), which are then expanded to the respective model sequences, and added to the search space. The search beam is pruned to keep only a little number of promising word sequences

### 2.2.5 Speech recognition using WFST

Weighted Finite State Transducers (WFST) offer a way to integrate phonetic modeling, lexicon and language model based on weighted acceptors and transducers.

#### Transducer composition

$$R = H \circ C \circ L \circ G$$

- First layer - Grammar (G): Models the possible word sequences to be recognized based on the language model. It is an acceptor, i.e, its input and output symbols are the same.
- Second layer - Lexicon (L): Takes phoneme symbols as input and produces words as output.
- Third layer - Context dependency (C): Models a transducer whose output is a sequence of phoneme symbols depending on the left and right phonetic context.
- Fourth layer - HMM (H): Models the actual HMM. It uses an emission distribution (or state) ID as input, which is used by the decoder to compute the actual emission probability, and outputs context-dependent phones.

### **Training and decoding**

Both training and decoding are based on the path within the compound WFST  $R$  that represents the state sequence of the underlying HMMs. Viterbi algorithm can be implemented straight forward only for small WFST since an unacceptably high large WFST results when longer utterances and pronunciation alternatives are considered. Therefore a modification of Viterbi algorithm, the time-synchronous Viterbi beam search, which is based on considering only a certain number of best paths, referred as the beam, is implemented. The pruning step is either implemented by considering a fixed number of paths, or by allowing paths that are within a certain likelihood range of the current best path.

### 3 ASR for multiple languages

A system that is designed to recognize one particular language at a time is denoted as monolingual system. Such system is trained solely using data from the target language to recognize. This kind of system was introduced in Sec 2. A multilingual system, in contrast to a monolingual, is capable to recognize several languages. In the following sections it is described the architecture of a multilingual ASR system and the resulting challenges for speech recognition. It is also discussed an efficient way deal with the enormous task of covering a small percentage of the word's languages by building speech recognition systems for multiple languages through model combination.

#### 3.1 Multilingual ASR systems

Integrating several monolingual recognizers with a front-end for language identification is a very basic approach to identify multiple languages. Ideally, this concept could accomplish the required task; however, storage requirements set a limit on this approximation. Combining different parameters yields to the concept of a multilingual engine, thereby several different languages could be identify while saving storage space. The concept of sharing is shown in the Fig. 7. Knowledge sharing can happen on three levels: the acoustic model, the pronunciation dictionary, and the language model. Fig. 7 presents the architecture when the acoustic model is the shared parameter in a multilingual system.

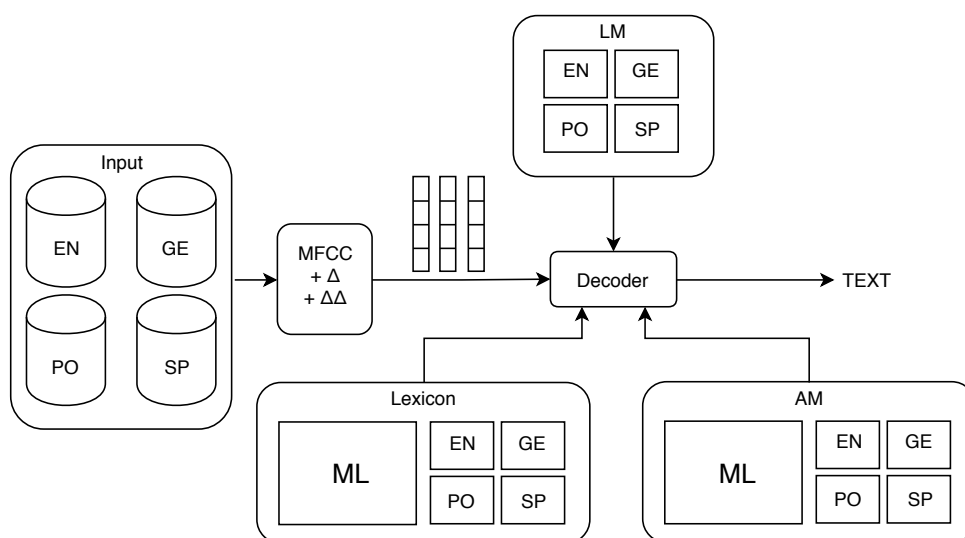


Figure 7 Multilingual ASR system

### 3.1.1 Lack of resources

Per language, about 10,000 utterances, or 100,000 words spoken by 100 different speakers, are said to be the minimal amount to guarantee a relatively robust LVCSR system [2]. Such large amount of required data explains why ASR technology has successfully been introduced in commercial system for resource-rich languages, e.g. English, Chinese, French; where there are adequate resources [25]. However, there are some languages denoted as under-resource languages where required resources to build a ASR system are not available. The term lack can be extended to at least one of the following: unique writing system, stable orthography, linguistic expertise, transcribed speech data, pronunciation dictionaries, vocabulary lists, electronic text resources, etc.

The lack of such data is one of the major challenges in the development process of speech recognizers that has also driven to the idea of sharing models between different languages. When the acoustic model is the target resource to share, the quantity and quality of the speech and text databases have to be balanced across languages. For an ideal multilingual acoustic model, balance alludes to

- Amount of text and audio data across languages.
- The audio data should cover a reasonable number of speakers with a representative distribution in demographics, such as gender, age, place of origin, and education.
- Audio material should fulfill some conditions to ensure quality material. Some of them are noise, microphone, soundcard.
- The amount of text data should allow reliable estimates for language modeling or grammar development.

### 3.1.2 Acoustic model combination

The acoustic model combination approach is based on the fact that the sounds produced across languages share a common acoustic space [25]. The usual mechanism to implement a combined acoustic model consists of defining a lexical unit set based on universal phones using either knowledge-based or data-driven approaches. Once the universal phone set is defined, the relationship between lexical units and acoustic feature observations is learned on language-independent data.

## 3.2 Language resources: Corpora and lexica

The benefits of the definition of a unified multilingual phoneme set become more and more convenient as the number of languages to cover grows. This is because the more languages covered, the more phonetic similarities across languages can be exploited [26]. The first step towards the design of the multilingual speech recognition presented in this work was then the identification of the different phonetic representations incorporated by the databases used. Afterwards, once the similarities across them were identified, the definition of a unified phoneme inventory that could equally work for all languages was defined.

### 3.2.1 TIMIT corpus

The TIMIT speech database is an acoustic-phonetic continuous speech corpus of American English. The speech material was exclusively collected from native speakers and it covers the major 8 dialects of the United States. TIMIT is intended to provide acoustic data and phonetic information for the evaluation and improvement of ASR systems. This corpus was created by Massachusetts Institute of Technology (MIT), Stanford Research Institute (SRI), and Texas Instruments (TI), and was sponsored by the Defense Advanced Research Projects Agency - Information Science and Technology Office (DARPA - ISTO). The phonetic alphabet used of the TIMIT corpus is sometimes called TIMITBET and is composed by 61 different symbols.

### 3.2.2 Wall Street Journal corpus

The DARPA Wall Street Journal corpus (WSJ) is a general-purpose English, large vocabulary, natural language, high perplexity corpus containing significant quantities of both speech data (400 hrs) and text data (47M words) [27]. Text materials were selected to provide training and test for 5k and 20k words. This corpus was sponsored by the Advance Research Projects Agency (ARPA) and Linguistic Data Consortium (LDC), and carried out by MIT, TI and Standford Research Institute (SRI) International. WSJ was collected in two phases, namely CSR-WSJ0 and CSR-WSJ1. These two phases, which differ from the year of data collection, determine how the proposed training, development and evaluation sets are predefined. The different noises that can be encounter in WSJ are made by cars, restaurants, streets, airport, and train. No phonetic lexicon was designed to be used exclusively by this database. Instead, the CMU-Arctic phoneset which is also known as CMUBET together with the Carnegie Mellon University (CMUdict) dictionary are used.

### 3.2.3 GlobalPhone database

GlobalPhone (GP) is, up to date, a 22 languages high-quality read speech and text database designed to be uniform across languages in terms of the amount of text and audio per language [2]. The aim of GP corpus is to provide transcribed speech data for the development and evaluation of LVCSR systems. Languages such as German, Portuguese and Spanish can be found among the languages covered by GP. The data acquisition was performed where the language is officially spoken. German was collected in Karlsruhe, Germany; Brazilian Portuguese in Porto Velho and São Paulo, Brazil; and Spanish in Heredia and San José, Costa Rica [28]. The read texts were selected from national newspaper articles from the 1995 to 1998 international political and economic news. The transcriptions are available in the original orthographic script, but additionally mapped into a romanized form. The romanized version of all transcripts is encoded in ASCII-7. The original orthographic scripts for German, Spanish and Portuguese is ISO-8859-2. The phonetic representation of each language the GP covers changes from one language to another. Therefore, the size of the phone inventory, as well as the used symbols are dependent of the language. The phone set of German, Portuguese and Spanish are composed of 41, 45, and 40 different symbols respectively.

### 3.3 Phonetic alphabets

Typically, ASR systems use linguistically motivated phones or phonemes as subword units. Therefore, different phone sets have been designed to cover the set of sounds in all languages. Examples of phone sets include the TIMITBET and CMUBET previously mentioned when the TIMIT and WSJ corpora were introduced. Because the target of this work is to build a multilingual system whose acoustic model can be shared across the implemented languages, a unified phone representation must be defined because of the variety of phone alphabets found in the used language resources. As stated in Sec. 3.1.2, the combination of different acoustic models requires a language-independent phone set definition.

#### 3.3.1 The international phonetic alphabet

The International Phonetic Alphabet (IPA) is an internationally used notational system for transcribing at phonetic level. It is based on the Roman alphabet, but also includes letters and additional symbols from other sources in order to cover the wide variety of sounds found in the languages of the world. Consonants and vowels are separated in the IPA chart for articulatory reasons. Vowels are organised into a chart referred as the vowel quadrant or vowel space. This chart is intended to roughly represent the physical space inside ones mouth. The left of the chart represents the mouth portion closer to the lips, and the right side is the back of the mouth. The top of the chart is the roof of the mouth and the bottom of the chart is the jaw.

The topmost table represent the basic consonants. Each column represents where in the vocal tract the constriction takes place, and each row represents how much constriction there is. Consonants come in pairs on the chart, and these pairs are based on vocal fold vibration [29].

#### Machine-readable extension: SAMPA and X-SAMPA

The necessity of writing the phonetic transcription encoded in such way that could be machine-readable led, in the late 1980s, to the development of the SAMPA. A 7-bit printable ASCII characters to represent the transcriptions of the IPA comprises SAMPA alphabet. The underlying principle of SAMPA was to select those IPA symbols which were conventionally used to represent phonemes in the major languages of the European Union. Since then IPA has been revised several times; consequently, extensions of SAMPA had to be implemented. The Extended SAMPA Phonetic Alphabet (X-SAMPA) then emerges to encompass the complete set of IPA conventions. A complete list of these two alphabets can be found in [30]. The mapping from IPA to either SAMPA or X-SAMPA is isomorphic so that a one-to-one transformation in both directions can be carried out.

#### 3.3.2 Alphabet for English

##### ARPABET

ARPAbet is an American English phonetic transcription code developed by ARPA as part of their Speech Understanding Project [31]. It uses ASCII symbols. There are two representations in ARPABET: one adopts only one character and includes lower-





num	IPA	Comp. repr.		num	IPA	Comp. repr.	
		1-Char	2-Chars			1-Char	2-Chars
1	i	i	IY	25	p	p	P
2	ɪ	I	IH	26	t	t	T
3	eɪ	e	EY	27	k	k	K
4	ɛ	E	EH	28	b	b	B
5	æ	@	AE	29	d	d	D
6	ɑ	a	AA	30	g	g	G
7	ʌ	A	AH	31	h	h	HH
8	ɔ	c	AO	32	f	f	F
9	oʊ	o	OW	33	θ	T	TH
10	ʊ	U	UH	34	s	s	S
11	u	u	UW	35	ʃ	S	SH
12	ə	x	AX	36	v	v	V
13	ɪ	X	IX	37	ð	D	DH
14	ɜ˞	R	ER	38	z	Z	Z
15	aʊ	W	AW	39	ʒ	Z	ZH
16	aɪ	Y	AY	40	tʃ	C	CH
17	ɔɪ	O	OY	41	dʒ	J	JH
19	w	w	W	42	l̩	L	EL
20	r	r	R	43	m̩	M	EM
21	l	l	L	44	n̩	N	EN
22	m	m	M	45	r	F	DX
23	n	n	N				
24	ɹ̩	G	NX				

**Table 1** ARPabet phoneme list and the X-SAMPA corresponding representation

case letters, while the second uses only upper-case letters. The two versions of the ARPabet are given in Table 1. Different lexica are derived from this phonetic code, e.g., CMUBET and TIMITBET.

## TIMITBET

The TIMIT phonetic representation, as mentioned before, is based on ARPABET phonetic alphabet. TIMIBET includes additional symbols to describe the enclosure and release parts of plosive sounds [32]. Because the target of this step is to define a phonetic alphabet that matches across all implemented languages, some modifications had to be made. Table 2 presents the resolved relationship between the TIMITBET and X-SAMPA conversion used in this work experiments. It can be seen that some characters, the ones followed by a superscript, were differently mapped of how they should conventionally be converted from the TIMITBET to X-SAMPA. The major motivation of such changes were imposed by CMUBET. The standardize mapping can be found in the footnotes.

<sup>1</sup>/em/ → m=

<sup>2</sup>/en/ → n=

<sup>3</sup>/ah/ → V

<sup>4</sup>/axr/ → @'

<sup>5</sup>/ix/ → 1

<sup>6</sup>/el/ → l=

num	TIMITBET	IPA	X-SAMPA	num	TIMITBET	IPA	X-SAMPA
1	aa	ɑ	A	24	em	ɱ	m <sup>1</sup>
2	ae	æ	{	25	en	ɳ	n <sup>2</sup>
3	ah	ʌ	@ <sup>3</sup>	26	f	f	f
4	ao	ɔ	O	27	g	g	g
5	aw	aʊ	aU	28	hh	h	h
6	ax	ə	@	29	jh	dʒ	dZ
7	axr	ə <sup>˘</sup>	3 <sup>4</sup>	30	k	k	k
8	ay	aɪ	aI	31	l	l	l
9	eh	ɛ	E	32	m	m	m
10	er	ɜ <sup>˘</sup>	3 <sup>˘</sup>	33	n	n	n
11	ey	eɪ	eI	34	ng	ŋ	N
12	ih	ɪ	I	35	p	p	p
13	ix	i	I <sup>5</sup>	36	r	r	r
14	iy	i	i	37	s	s	s
15	ow	oʊ	oU	38	sh	ʃ	S
16	oy	ɔɪ	OI	39	t	t	t
17	uh	ʊ	U	40	th	θ	T
18	uw	u	u	41	v	v	v
19	b	b	b	42	w	w	w
20	ch	tʃ	tS	43	y	j	j
21	d	d	d	44	z	z	z
22	dh	ð	D	45	zh	ʒ	Z
23	el	ɪ	I <sup>6</sup>				

**Table 2** Mapping used from the TIMIT phoneme list to X-SAMPA

### CMUBET

This CMUBET alphabet is also based on the ARPABET American English phonetic transcription. CMUBET do not specify additional symbols in its inventory as TIMIT. Instead, it include less phones that the standardize phone inventory defined by ARPABET . The difference between TIMIT and CMUBET, as well as the need of unifying the phone representation between these two dictionaries led to implementing some modifications. The one-to-one mapping from CMUBET to X-SAMPA is presented in Table 4. Taking the word "beautiful" illustrates the transcription encountered mismatch between TIMIT and CMUDICT lexica.

	CMUDICT	TIMITBET
Lexica transcription	B Y UW1 T AH0 F AH0 L	b y uw1 t ih f el
X-SAMPA standard conversion	b j u t V f V l	b j y t I f I l=

### 3.3.3 Alphabet for German

The conversion from the GlobalPhone phonetic representation to X-SAMPA was based on the documentation provided by the database. The definition of each phonetic symbol used in GP transcription is presented in term of the IPA phonetic alphabet [33]. The 41 phone set implemented by GP covers 46035 dictionary entries. Table 4 presents the definition of the defined X-SAMPA representation for German.

num	CMUBET	IPA	XSAMPA	num	CMUBET	IPA	X-SAMPA
1	aa	ɑ	A	21	g	g	g
2	ae	æ	{	22	hh	h	h
3	ah	ʌ	@ <sup>7</sup>	23	jh	dʒ	dZ
4	ao	ɔ	O	24	k	k	k
5	aw	aʊ	aU	25	l	l	l
6	ay	aɪ	aI	26	m	m	m
7	eh	ɛ	E	27	n	n	n
8	er	ɜ <sup>v</sup>	3 <sup>v</sup>	28	ng	ŋ	N
9	ey	eɪ	eI	29	p	p	p
10	ih	ɪ	I	30	r	r	r
11	iy	i	i	31	s	s	s
12	ow	oʊ	oU	32	sh	ʃ	S
13	oy	oɪ	oI	33	t	t	t
14	uh	ʊ	U	34	th	θ	T
15	uw	u	u	Z	v	v	v
16	b	b	b	36	w	w	w
17	ch	tʃ	tS	37	y	j	j
18	d	d	d	38	z	z	z
19	dh	ð	D	39	zh	ʒ	Z
20	f	f	f				

**Table 3** Mapping used from the CMUDICT phoneme list to X-SAMPA

### 3.3.4 Alphabet for Portuguese

The X-SAMPA definition for Portuguese was unclear at the beginning because the ambiguity found in the GP specifications. According to the documentation certain vowels can be mapped to the different IPA symbols. This problem is clear depicted in Fig. 9. It can be observed, in the case of /E/ GP phonetic unit, that such symbol could be mapped as either /e/, /ə/ or /ɛ/. The same inconvenient can be also observed for /O/ since it can be mapped as /o/ or /ɔ/. It was finally decided to make use of the X-SAMPA phonetic symbols for /e/ and /o/ IPA representation. The total 45 phone set is used by GP to cover 58878 dictionary entries. Table. 5 shows the complete mapping of the GP phonetic alphabet to X-SAMPA.

### 3.3.5 Alphabet for Spanish

The Spanish language uses a phonetic alphabet composed by 40 phones. Because Spanish has a straightforward grapheme-to-phoneme relationship, the phonetic representation presented by GP was automatically created by a set of grapheme-to-phoneme mapping rules [35]. The GP phoneme alphabet for Spanish covers 33960 words occurring in the transcription. Table. 6 presents the conversion implemented in the developed multilingual system.

### 3.3.6 Sharing phones

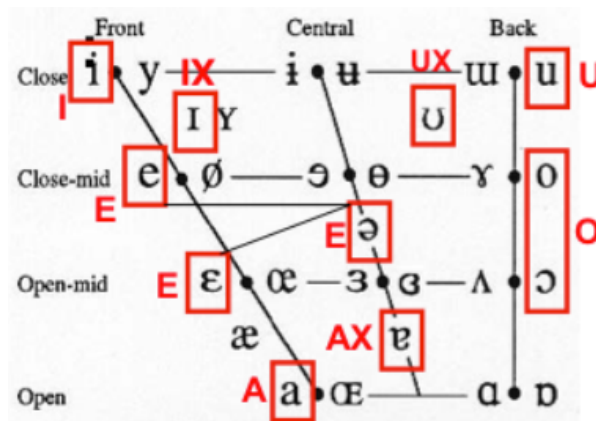
Table. 7 and 8 present a review of the achieved representation after joining the pronunciation of English, German, Portuguese and Spanish. In one hand, It can be noticed

---

<sup>7</sup>/ah/ → V

num	GP Dict.	IPA	XSAMPA	num	GP Dict.	IPA	X-SAMPA
1	a	a	a	22	C	ç	C
2	ae	ɜ	E	23	d	d	d
3	atu	ɐ	6	24	g	g	g
4	e	e	e	25	f	f	f
5	etu	ə	@	26	h	h	h
6	i	i	i	27	j	j	j
7	o	o	o	28	k	k	k
8	oe	œ	9	29	l	l	l
9	u	u	u	30	m	m	m
10	ue	y	y	31	n	n	n
11	aI	aɪ	aI	32	ng	ŋ	N
12	aU	aʊ	aU	33	p	p	p
13	eU	ɔy	OY	34	r	r	r
14	al	a:	a:	35	s	s	s
15	el	e:	e:	36	S	ʃ	S
16	il	i:	i:	37	t	t	t
17	oel	ø:	2:	38	ts	ts	ts
18	ol	o:	o:	39	v	v	v
19	uel	y:	y:	40	x	x	x
20	ul	u:	u:	41	z	z	z
21	b	b	b				

**Table 4** Mapping used from the GP-German phoneme list to X-SAMPA



**Figure 9** Definition of vowel representation defined by GP for Portuguese. Taken from [34]

that most of the consonants across these four languages are shared. On the other hand, vowels significantly vary from one to another language. Therefore, It can be said then that the multilingual system is more likely to accurately recognize consonants since the amount acoustic training material of such phones is higher than the material of vowels. Additionally, a general review is also presented in Tables 9 and 10

### 3.4 Language models

An N-gram language model, presented in Sec. 2.2.3 represents a probability distribution over words  $w$ , conditioned on  $(N - 1)$ -tuples of preceding words or histories  $h$ .

num	GP Dict.	IPA	XSAMPA	num	GP Dict.	IPA	X-SAMPA
1	a	a	a	21	w	v	P
2	a+	a'	a"	22	t	t	t
3	e	e	e	23	d	d	d
4	e+	e'	e"	24	n	n	n
5	i	i	i	25	r	r	r
6	i+	i'	i"	26	rf	r	4
7	o	o	o	27	T	θ	T
8	o+	o'	o"	28	D	ð	D
9	u	u	u	29	s	s	s
10	u+	u'	u"	30	z	z	z
11	ai	ai	ai	31	l	l	l
12	au	au	au	32	n~	ɲ	J
13	ei	ei	ei	33	j	j	j
14	eu	eu	eu	34	L	ʎ	L
15	oi	oi	oi	35	k	k	k
16	p	p	p	36	g	g	g
17	b	b	b	37	x	x	x
18	m	m	m	38	G	ɣ	G
19	V	β	B	39	ng	ŋ	N\
20	f	f	f	40	tS	tʃ	tS

**Table 5** Mapping used from the GP-Portuguese phoneme list to X-SAMPA

num	GP.	IPA	XSAMPA	num	GP.	IPA	X-SAMPA
1	a	a	a	21	w	v	P
2	a+	a'	a"	22	t	t	t
3	e	e	e	23	d	d	d
4	e+	e'	e"	24	n	n	n
5	i	i	i	25	r	r	r
6	i+	i'	i"	26	rf	r	4
7	o	o	o	27	T	θ	T
8	o+	o'	o"	28	D	ð	D
9	u	u	u	29	s	s	s
10	u+	u'	u"	30	z	z	z
11	ai	ai	ai	31	l	l	l
12	au	au	au	32	n~	ɲ	J
13	ei	ei	ei	33	j	j	j
14	eu	eu	eu	34	L	ʎ	L
15	oi	oi	oi	35	k	k	k
16	p	p	p	36	g	g	g
17	b	b	b	37	x	x	x
18	m	m	m	38	G	ɣ	G
19	V	β	B	39	ng	ŋ	N\
20	f	f	f	40	tS	tʃ	tS

**Table 6** Mapping used from the GP-Spanish phoneme list to X-SAMPA

Vowels							
num	IPA	X-SAMPA	Languages	num	IPA	X-SAMPA	Languages
1	ɑ	A	EN	26	o	o	GE, PO, SP
2	ʌ	@	EN, GE	27	oː	oː	GE
3	øː	2ː	GE	28	oʰ	o"	PO, SP
4	ɐ	6	GE, PO	29	õ	o~	PO
5	a	a	GE, PO, SP	30	õʰ	o"~	PO
6	aː	aː	GE	31	ʊ	U	EN, PO
7	aʰ	a"	PO, SP	32	u	u	EN, GE, PO, SP
8	ã	a~	PO	33	uː	uː	GE
9	ãʰ	a"~	PO	34	uʰ	u"	PO, SP
10	ɛ	E	EN, GE	35	ũ	u~	PO
11	ɜ̃	3˘	EN	36	ũʰ	u"~	PO
12	e	e	GE, PO, SP	37	æ	{	EN
13	eː	eː	GE	38	aɪ	aI	EN, GE
14	eʰ	e"	PO, SP	39	ai	ai	SP
15	ẽ	e~	PO	40	aʊ	aU	EN, GE
16	ẽʰ	e"~	PO	41	au	au	SP
17	ɪ	I	EN, PO	42	eɪ	eI	EN
18	i	i	EN, GE, PO, SP	43	ei	ei	SP
19	iː	iː	GE	44	eu	eu	SP
20	iʰ	i"	PO, SP	45	ɔɻ	OY	GE
21	ĩ	i~	PO	46	œ	9	GE
22	ĩʰ	i"~	PO	47	r	4	SP
23	y	y	GE	48	oi	oi	SP
24	yː	yː	GE	49	ou	oU	EN
25	ɔ	O	EN				

**Table 7** Shared vowels after the implemented conversion

Consonants							
num	IPA	X-SAMPA	Languages	num	IPA	X-SAMPA	Languages
1	β	B	SP	20	ŋ	N	EN, GE
2	b	b	EN, GE, PO, SP	21	v	P	SP
3	ç	C	GE	22	p	p	EN, GE, PO, SP
4	ð	D	EN, SP	23	ɾ	R	PO
5	d	d	EN, GE, PO, SP	24	r	r	EN, GE, PO, SP
6	dʲ	dʰ	PO	25	ʃ	S	EN, GE, PO
7	dʒ	dZ	EN	26	s	s	EN, GE, PO, SP
8	f	f	EN, GE, PO, SP	27	θ	T	EN, SP
9	ɣ	G	SP	28	t	t	EN, GE, PO, SP
10	g	g	EN, GE, PO, SP	29	tʲ	tʰ	PO
11	h	h	EN, GE	30	tʃ	tS	EN, SP
12	ɲ	J	PO, SP	31	ts	ts	GE
13	j	j	EN, GE, SP	32	v	v	EN, GE, PO
14	k	k	EN, GE, PO, SP	33	w	w	EN
15	ʎ	L	PO, SP	34	x	x	GE, SP
16	l	l	EN, GE, PO, SP	35	ʒ	Z	EN
17	m	m	EN, GE, PO, SP	36	z	z	EN, GE, PO, SP
18	ɲ	N\	SP	27	r	4	SP
19	n	n	EN, GE, PO, SP				

**Table 8** Shared vowels after the implemented conversion

Languages	Num. of phones	Shared phones	Unique phones
EN	39	29	10
GE	41	29	12
PO	43	30	13
SP	40	30	10

**Table 9** Overview of the total shared phones after the X-SAMPA conversion.

Languages	Num. of phones	Shared phones	Unique phones
Vowels			
EN	15	8	7
GE	20	10	10
PO	23	13	10
SP	15	10	5
Consonants			
EN	24	21	3
GE	21	19	2
PO	20	17	3
SP	25	20	5

**Table 10** Review of the shared phones for vowels and consonants

Because such probabilities indirectly encode the relevant aspects of a language, e.g. the syntax, grammar rules, etc.; LMs have to be independently defined per language. The multilingual system developed in this work covers four languages; consequently, it was necessary to count with at least four different language models that describe each of them separately. The language model used for English was included in the WSJ corpus and was developed by the MIT Lincoln Laboratory; whereas the language models corresponding to the GP languages were obtained from the Internet. These are public distributed and can be found at [36]. Despite they all were obtained from different resources, they were stored in the same text file, namely ARPA-MIT format. It is widely used because most of the language model toolkits support this format.

**Wall Street Journal** WSJ offers a variety of meaningful statistical language models distributions depending on the WSJ development phase, this refers to the WSJ0 and WSJ1 distributions. Some difference between them reside in the vocabulary coverage, n-gram order, insertion of verbalized punctuation pronunciation, etc. The chosen language model in this work corresponds to the WSJ1 distribution, open vocabulary back-off trigram model. More details concerning the selected language model are shown in Table 11

**Globalphone** The language models of GP data base were built by RLAT. This is a web-based interface which aims to reduce the human effort involved in building speech processing systems for new languages [37]. One website per language was chosen to collect text data; the removal of HTML tags, code fragments, and empty lines were remove and then used to create the language models. The process consisted in gathering information on a daily basis and building different LMs using the daily crawled data. The final language models are a result by a linear interpolation of all daily built language models. The weights were computed by using SRI language toolkit. A detailed description of the most relevant specifications per language model used can be found



Lang	3gram	3gram-prune	PPL	OOV (%)	Vocab
EN	3153527	709089	109	1.5	20k
GE	990676	116824	672	0.3	38k
PO	2625824	347190	58	9.9	62k
SP	15986	12037	154	0.1	19k

**Table 11** Language model specifications.

in Table 11

## 4 Experimental part

### 4.1 Language resources

Different corpora were used to build the multilingual system presented in this chapter. Globalphone and WSJ are the two data bases the system is made up with. Globalphone contributes with German (GE), Spanish (SP) and Portuguese (PO) languages; while WSJ with English (EN) language data. In Sec. 3.3 it was described the phonetic alphabet each of these corpora implement by default and what conversion was resolved to unified the phonetic representation. This section is addressed to describe the different sets for the training and evaluation tasks.

**Wall Street Journal** The WSJ training and test set are recorded with a close and a far talking microphone. Speech recorded by the far-talking microphone is mainly used to research the effect of changing channel characteristics which is not the focus of this thesis. Hence, the only speech data used is the close talking microphone. Such audio files are identified by the .wv1 extension. The training, evaluation and development set structures proposed by the original WSJ recipe can be found in Table 12 and 13

	WSJ0	WSJ1	WSJ0+WSJ1
Speakers	84	200	284
Sentences	7240	30276	37516
Words	132472	511527	643999

**Table 12** WSJ data structure.

Corpus	Development data				Evaluation data			
	Dev92		Dev93		Nov92		Nov83	
	5k	20k	5k	20k	5k	20k	5k	20k
Speakers	10	10	10	10	8	8	10	10
Sentences	410	403	513	503	330	333	215	213
Words	6780	6742	8639	8235	5353	5643	3854	3448

**Table 13** WSJ training corpus

**GlobalPhone** GlobalPhone training, development, and evaluation sets are split up at an 80 : 10 : 10 ratio in such a way that no speaker appears in more than one group and no article is read by two speakers from different groups. Table 14 shows the data distribution of each set for German, Portuguese and Spanish according to the original recipe.

**Merge of databases** To have approximately the same contribution per language was the design parameter that determined the size of each set to implement in the multilingual system. On the basis of this requirement, a redefinition of the training, evaluation

	Lang	Training	Evaluation	Development
Speakers	GE	65	6	6
	PO	86	7	8
	SP	82	8	10
	Total	233	21	24
Utterances	GE	8185	826	1073
	PO	8928	694	648
	SP	5425	564	677
	Total	22538	2084	2398
Words	GE	115617	11959	15387
	PO	177779	15540	13030
	SP	138033	14426	19098
	Total	431429	41925	47515

**Table 14** GP data structure for German (GE), Portuguese (PO) and Spanish (SP)

	Lang	Training	Evaluation	Development
Speakers	EN	89	15	10
	GP	233	21	24
	Total	322	36	34
Utterances	EN	8000	800	1000
	GP	22548	2084	2398
	Total	30538	2834	3398
Words	EN	144323	13404	16081
	GP	431429	41925	47515
	Total	575752	55329	63596

**Table 15** Multilingual system set structure by languages

and development sets for English had to be done. This is because the training set define by the WSJ recipe includes approximately six times more words than each language in the GP data base. The final redefinition of each English set can be observed in Table 15 as well as the rest of the details that totally described the data to be used in the multilingual ASR development stage. No changes were made in the GP sets.

## 4.2 Evaluation metric

The performance of a speech recognition system is typically measured by comparing the recognized word sequence (hypothesis) with the reference word sequence that was obtained by manual transcription [38]. In CSR systems there are three types of errors:

- Insertion: An extra word is added to the the recognized word sequence
- Substitution: A correct word in the word sequence is replaced by an incorrect word
- Deletion: A correct word in the word sequence is omitted.

To determine the recognition accuracy, the recognized word string has to be align against the correct word string, and then compute the number of the corrected, substituted, deleted and inserted words. This alignment can be obtained using a dynamic

programming algorithm [39]. Word error rate (WER) is the metric of first choice for determining the quality of automatically derived speech transcriptions. If there are  $N$  words in the reference transcript, and alignment with the speech recognition output results in  $S$  substitutions,  $D$  deletions, and  $I$  insertions the word error rate can be readily calculated as

$$WER = \frac{S + D + I}{N} \times 100\% \quad (7)$$

### 4.3 Implementation of ASR in KALDI

#### 4.3.1 KALDI framework

KALDI is an open-source toolkit for speech recognition research written in C++ [40]. The standard workflow provides a robust starting point. Recipes, which are templates for training acoustic models on a given speech data, are divided into different stages that correspond to the feature extraction, training, evaluation and decoding. KALDI is theoretically grounded in the finite states machines (recall Sec. 2.2.5). The basic operations that support this framework are:

##### **Composition**

Transducer operation for combining different levels of representation.

##### **Determinization**

Transformation of a non-deterministic weighted automaton into an equivalent deterministic automaton. This operation is applied to remove redundancy, thereby reducing the time and space needed to process the string.

##### **Minimization**

This operator outputs an automaton  $B$  which has the least number of states and the least number of transitions among all deterministic weighted automata equivalent to the input  $A$ .

##### **Epsilon removal**

Operation that removes epsilon transitions from transducer. If an epsilon is used as output and input label, transitions do not produce either any or output symbol, so the removal can be performed without losing function of the transducer.

#### 4.3.2 Setup of recipes stages

To build the monolingual and multilingual ASR systems, the standard s5 recipe structure was implemented. The modifications made to the original recipe are also explained in the following section

##### **Stage 0: Data & lexicon & language preparation**

The first step to complete according to the presented ASR architecture in Fig. 1 corresponds to the front-end module or feature extraction processing. However, in practice an additional step is required. Stage 0 prepares the data into a common standardized

	Disambiguation sym.		Silence phones		Optional silence	
	Token	Phone	Token	Phone	Token	Phone
EN	<SPOKEN NOISE>	spn	<SPOKEN NOISE> <UNK> <NOISE> !SIL	spn spn spn sil	!SIL	sil
GP	<unk>	spn	<unk> !SIL	spn sil	!SIL	sil

**Table 16** Assignment of disambiguation symbols, silence and nonsilence phones for EN and GP languages

format. The goal after the execution of this stage is

- Definition of training, evaluation and development sets. The arrangement of these three sets is shown in Table 15.
- Creation of L and G files based on the input lexicon dictionary and language model respectively.
- Definition of disambiguation symbols, optional and silence phones Although all languages were assigned the same representation of the disambiguation symbol and optional/silence phones, their token vary between EN and GP languages. Table 16 presents the tokens and representation used.
- Definition of the silence and non silence number of states. These two parameters were setup using 3 and 5 number of states respectively. KALDI makes use of the CD phones when the option `--position-dependent-phones true` is used in the script `prepare_lang.sh`.
- Definition of the pruning threshold. As mentioned in Sec. 3.4, the language model probabilities are specified in a text file. Therefore, KALDI during this stage recalculates some probabilities from the non-pruned language model files. Sec. 2.2.5. The chosen threshold is  $10^7$ .

This stage outputs multiple files, all necessary for execution of further steps. Moreover, most of them are automatically and successfully generated if correct input data is set. More details can be found in the official webiste of KALDI [40].

### Implemented modifications

1. Initially all databases work with their own complete audio resources. Consequently, the first implemented change was addressed to resize the training, evaluation and development set of English. To perform this task the data preparation script `local/EN/cstr_wsj_data_prep.sh` was modified. The added lines are intended to make sure that the new dimensions are equal or less than the maximum amount of audio resources each database counts with. An example of this instruction is shown in the following piece of code.

**Listing 4.1** Re-definition of the training set for EN.

```
if [ 'wc -l < train_si284.flist' -gt $train ];then
    head -$train train_si284.flist > train_resize.flist
fi
```

2. The WSJ does not include any lexicon in the corpus, thus CMUdict (version 0.7a) was utilized as base dictionary (recall Section 3.2.2). The developed ASR system in this thesis was not meant to handle lexical stress markers, thus such symbols were removed from the original lexicon (numbers 0, 1 and 2). The mapping from the standard CMUdict pronunciation to X-SAMPA representation is carried out by an additional script which is not included in the standard WSJ recipe. This script is found in `local/lexicon_convert.pl`.
3. The GP data base provides, per language, the possibility customize the characters to use during the acoustic model training by the default files located in `local/GP/gp_norm_dict_().pl` and `conf/GP/xsampa_map/()` – the parenthesis represent GE, PO, and SP. In addition to this, there is an additional script supplied by the standard recipe, `local/GP/gp_norm_trans_().pl`, which basically converts the input text into UTF-8 encoding. Unfortunately a correct mapping was not obtained by the example scripts available in KALDI distribution; as a result an independent conversion script was developed. The scripts for GE, PO and SP were written altogether with the CMUdict mapping in the same file `local/lexicon_convert.pl`. The word "rítmica" taken from Portuguese language is used in the following example to illustrate the incorrect conversion. The correct X-SAMPA representation of the character RR (IPA: ʀ) is R, but the incorrect representation rr was obtained by the GP scripts:

```
Dictionary: {{RR WB} I+ TJ I M I K {AX WB}}
GP PO script: rr i" t' i m i k 6
lexicon_convert.pl: R i" t' i m i k 6
```

4. Attention has to be paid when running the script `local/GP/gp_data_prep.sh` because it internally runs another script, `local/GP/gp_convert_audio.sh`, which requires availability of the tool `sox` from the `path.sh`. It is recommended then to ensure that `sox` is ready to use somewhere in the recipe, the local directory was chosen in developed system. To set this tool up in the recipe the following lines, that can be obtained from the file `path.sh`, have to be run:

```
$PWD/tools/shorten-3.6.1/bin
SOX_BIN='pwd'/tools/sox-14.3.2/bin
```

5. Because the recognition task does not mix different languages up, the LMs do not have to be merged as the acoustic data. However; when the LM is adapted to the multilingual system, the pronunciation dictionary that covers all possible words to recognize across the different languages has to be used. In other words, when a LM is adapted to a given dictionary, KALDI employs certain numeration to identify the words and phones covered by the lexicon; so if a different numeration is used at any stage of the recognition task the system will automatically collide yielding incorrect results. This can be seen in the script `local/ML/ml_format_data.sh` the commands to generate the grammar G: `arpa2fst` are fed with the multilingual

System	Alignment	Training
Mono	MFCCs <code>steps/train_mono.sh</code>	-
Tri1	Align delta-based triphones <code>steps/align_si.sh</code>	Train delta + delta-delta triphones <code>steps/train_deltas.sh</code>
Tri2	Align delta + delta-delta triphones <code>steps/align_si.sh</code>	Train LDA-MLLT triphones <code>steps/train_lda_mllt.sh</code>
Tri3	Align LDA-MLLT triphones with fMLLR <code>steps/align_fmllr.sh</code>	Train SAT triphones <code>steps/train_sat.sh</code>

**Table 17** Overview of the training and alignment algorithms of the built Multilingual speech recognition system

dictionary.

### Stage 1: MFCC Feature Extration & CMVN

The system up to this stage counts with the information enough concerning what acoustic recourses are going to be parametrized. The algorithm implemented to perform the feature extraction corresponds to the built-in script `steps/make_mfcc.sh`. The settings of this stage are

- Frame length: 25 ms
- Time shifting: 10 ms
- Window: Hamming
- Number of Mel frequency bins: 23
- Number of cepstra in MFCC computation (including C0): 13
- Sampling frequency: 16 KHz
- The CMVN is computed per speaker by `steps/compute_cmvn_stats.sh`.

Although in Fig. 3 it is specified that the dynamic features delta ( $\Delta$ ) and delta-delta ( $\Delta\Delta$ ) are intrinsically incorporated in the MFCC processing, KALDI estimates these derivatives as an independent process to MFCC feature estimation.

### Stage 2: Acoustic model training

Different acoustic modeling units, decoding phases and training algorithms were used to build the multilingual ASR system presented in this thesis. The majority of them are based on triphone models since they represent a phoneme variant in the context of two other left and right phonemes. (discussed in Sec. 2.2.1).

KALDI optimizes the estimation of the acoustic model parameters by cyclically repeating an alignment and training phase in every acoustic training step. By aligning the audio to the reference transcript with the most current acoustic model in each training step, additional algorithms can then be used to improve or refine the parameters of the model. A review of the training procedure with the used respective scripts per model are presented in Table 17. Despite four different systems were implemented in this multilingual system (mono, tri1, tri2, tri3), KALDI offers the possibility to include further processing, e.g. Subspace gaussian mixture model adaptation and Maximum mutual information.

**Implemented modifications**

1. An additional level to the standard KALDI directory structure was added. This means that the output files had to be redirected to the either monolingual or multilingual system. The structure is then `exp/{EN,GE,ML,PO,SP}`, to store the recognition data, `mfcc/{EN,GE,ML, PO,SP}`, and lastly `data/{EN,GE,ML,PO,SP}` to store the acoustic features and data preparation scripts respectively. The addition of these directories, implemented only to avoid confusion between les, makes the addition to the language code to all paths called in the recipe.
2. The scripts presented in Table 17 include additional input arguments such as the number of leaves, or HMM states, or the number of Gaussians. Theoretically one HMM state could be assigned per phone, but that would not be convenient because phonemes vary considerably depending on if their position within a word (beginning, middle or end, discussed in Sec. 2.2.1). The numbers will largely depend on the amount of data, number of phonetic questions, and goal of the model. In Section ?? it is shown that numbers increase as the acoustic model is refined with further training algorithms.
3. The tri2 system of the standard GP recipe is obtained by calculating SAT directly from the dynamic features  $\Delta$  and  $\Delta\Delta$  (tri1). Despite this procedure is not incorrect whatsoever, it does not fit in the proposed systems described in Table 17. For this reason, the tri2 system implemented in the original GP recipe was replaced by the pertinent processing of LDA + MLLT; and default tri2 was renamed as tri3 taking as source to align the results from the new tri2 system.

**Stage 3: Making graphs**

The script `utils/mkgraph.sh` was the only script implemented to make the  $H \circ C \circ L \circ G$  final graph. This script creates a fully expanded decoding graph that represents the LM, pronunciation dictionary, context-dependency, and HMM structure. The output is a finite state transducer that has word-IDs on the output, and pdf-IDs on the input (these are indexes that resolve to Gaussian Mixture Models) [41]. No changes were made in this section.

**Stage 4: Decoding**

The last stage of the ASR system corresponds to generate the most likely word sequence given a model and an utterance. The script `decode.sh` is used for the majority of the acoustic models (mono, tri1, and tri2); whereas the system tri3 employs the script `decode_fmllr.sh`.

**Implemented modifications** The script which assigns the score in the decoding phase is usually found in the directory `local/` based on the standard KALDI recipe structure. Now, considering that an additional directory level was added to the file structure, it was necessary to provide to the both decoding scripts additional information about the path where the score file is located, and also what score file must be used depending on the language to recognize. To include this information, an additional input argument was added to each decoding script. This explains why all the decoding files are run from



the directory `local` instead of scripts found in `steps`. The following snippet of code is taken from `local/decode.sh` which illustrate the modifications just mentioned.

**Listing 4.2** Inclusion of the language variable (`$lang`) to find the proper score file

```
[ ! -x local/$lang/score.sh ] && \
  echo "Not scoring because local/$lang/score.sh
  does not exist or not executable." && exit 1;
local/$lang/score.sh --cmd "$cmd" $scoring_opts \
$data $graphdir $dir || { echo "$0: Scoring failed. \
(ignore by '--skip-scoring true')"; exit 1; }
fi
```

### Arrangement of steps to run the multilingual system

The system built in this work offers the possibility to perform the recognition task based on either a monolingual or multilingual system with a shared acoustic model. To run one of these systems it is necessary to run the script `run.sh` according to the following input command order:

```
run.sh <no. stage> <acoustic model> <language model>
```

- Number of stage: The assignation of each stage correspond to the same numeration previously presented.
- Acoustic model: Up to date, five options can be used as second input argument. They are: EN, GE, ML, PO, SP. When a different argument from ML is introduced, the script will utilize the monolingual acoustic model selected, and therefore the obtained results will be those obtained by a monolingual system with no contribution of any other language. In the other hand, when ML is chosen a third input argument has to be specified so that the system identifies what language model must be employed. An exception to this is when the stage 2 is run. This is because to specify the language to recognize is not needed when training the acoustic models.
- Language mode: As mentioned before, this option has to be specified when the multilingual acoustic model is used, i.e., when ML is introduced as second input argument. The language model defines what language is going to be used to build the HCLG graph in stage 3 and decode them in stage 4.

## 4.4 Results and discussions

The aim of this work is to develop a multilingual ASR system whose output for the implemented languages is as close as possible to the input word sequence. The following methodology was followed in order to ascertain a low WER across the covered languages. The results presented in this section are then based on these steps

- Preliminary test. A initial test was needed to perform in order to ensure that the resolved phoneme set did not produce incoherent results.
- Evaluation of multilingual systems. This means that the performance of each ASR system is evaluated individually when no contribution of other languages is presented.
- Integration of all languages into one single multilingual system whose acoustic model is shared.

#### 4.4.1 Preliminary test

The need to evaluate and verify the accuracy of the resolved mapping for English; and to make certain that the implemented modifications did not interfere with the system performance led to building an ASR system using the TIMIT data base as first step. Table 18 shows the number of speakers, utterances, and phones the training, and test sets comprise defined by the standard KALDI recipe for continuous speech recognition task - s5.

	Train	Eval
Speakers	462	168
Utterances	4620	1680
Words	39699	14518

**Table 18** Assigination of the train and evaluation tests in TIMIT database

Some advantages of testing the degree of correctness of the determined mapping using this corpus are related to the database size and literature availability. The size of this corpus was convenient in this evaluation phase since the computational time to carry out different experiments is not long. Additionally, the disposal of multiple literature resources ease the comparison between the typical WER values obtained by its corpus. The obtained WER results after the X-SAMPA representation are presented below in Table 19. The reference WER value presented in the last column was obtained by a HMM/GMM system whose total number of phone has been folded to a total of 39 phones [42]. The difference between these two results may stem from the number of data used during the training step. The system where the reference comes from did not use the complete training set for such purpose, instead just 2205 utterances were used. Nevertheless, the obtained result endorse the defined X-SAMPA conversion.

	Mono	Tri1	Tri2	Tri3	Ref
WER	60.70	55.98	54.78	53.04	61.5

**Table 19** Obtained WER for multilingual system based on TIMIT database

#### 4.4.2 Monolingual ASR systems

The second step described in the methodology corresponds to test each database independently from the others. This stage has two main objectives

- Ensure coherent results before merging all recipes. Getting consistent results in this stage also indirectly guarantees that the additional directory levels to the standard KALDI structure were correctly implemented.
- Evaluate the impact that the language mode and speaker sets have on the final WER

In Section 1, it was mentioned that the architecture of the back-end processing consists of three different resources. Two out of those three, the dictionary and the language model, are independent of the feature observation. Since obtaining the highest possible WER was pursued in this thesis, to asses what resources offer the best recognition had to be done. The different parameters to evaluate were the available language models and speaker sets. Different language models, 3-grams together with their pruned version; as

well as development and evaluation speaker sets were tested before proceeding with the multilingual ASR system. Fig. 11 presents the obtained WERs per system when varying the language model and sets. The same results can also be found in Table. 20

		Mono		Tri1		Tri2		Tri3	
		Tg	Tgpr	Tg	Tgpr	Tg	Tgpr	Tg	Tgpr
EN	Eval	19.29	20.87	9.56	10.22	8.16	8.38	6.79	7.49
	Dev	31.31	32.70	15.97	16.94	13.86	14.87	11.41	12.30
GE	Eval	42.54	46.98	23.53	25.64	21.76	24.20	17.62	19.19
	Dev	24.92	26.66	14.72	16.26	13.65	15.14	10.57	11.59
PO	Eval	38.49	54.89	26.99	32.92	26.10	30.32	24.86	28.93
	Dev	41.39	49.40	29.09	32.34	27.41	30.13	25.99	28.46
SP	Eval	25.22	26.45	12.84	13.27	11.58	11.96	10.16	10.54
	Dev	39.11	39.99	23.55	24.04	21.36	21.76	17.77	18.17

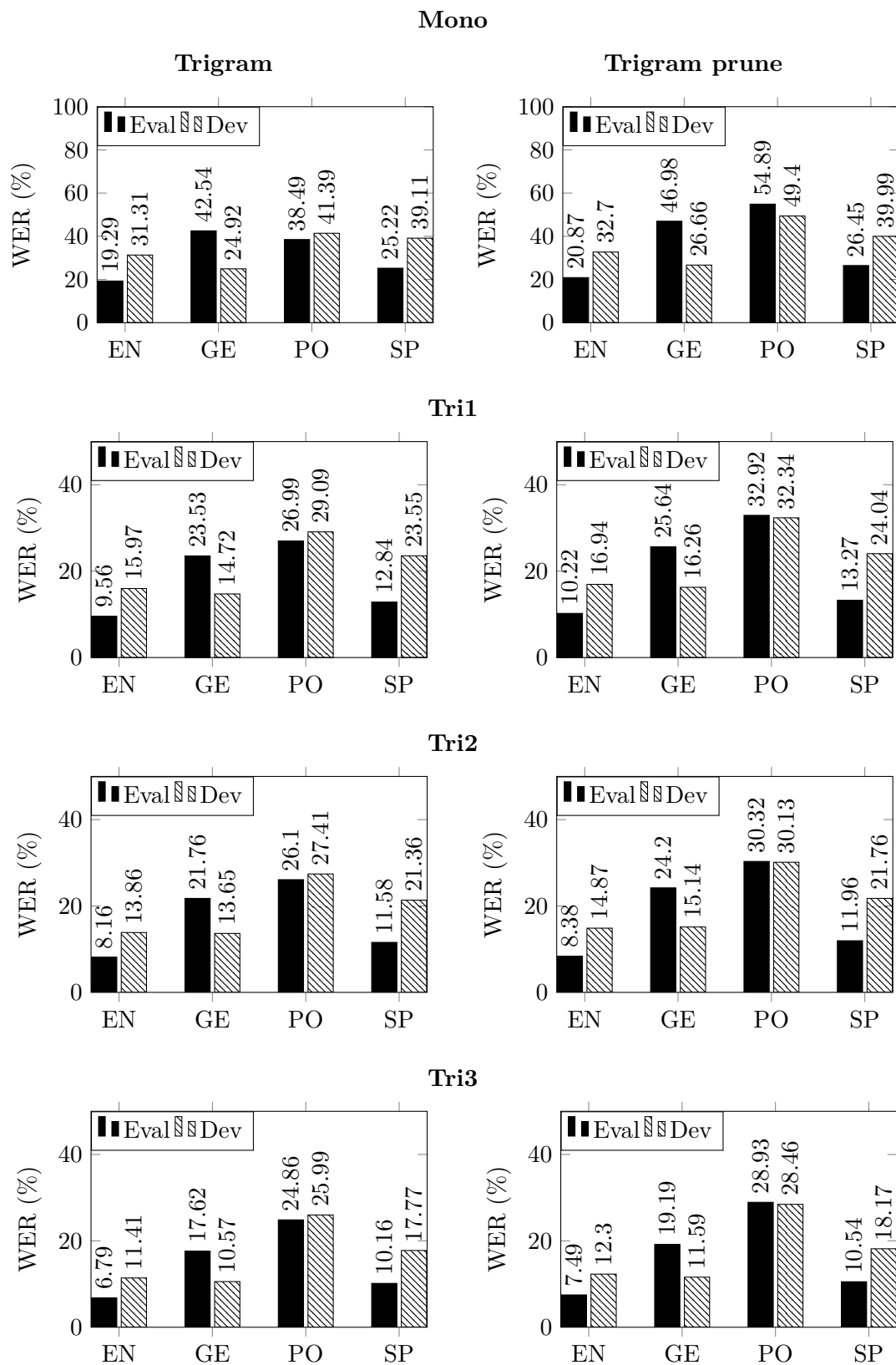
**Table 20** WER results when evaluation set and language model change. Results presented are obtained from monolingual systems.

From the results presented in Fig. 11 and Table. 20, the following ideas can be concluded:

- WER is not significantly affected when the language model changes from trigram to trigram prune, and additionally it could be say that the small changes are somehow predictable. In other words, if it is of interest to know what WER can be obtained using a pruned trigram model, an approximation can be estimated from the complete language model. This can also be declared in a reciprocal way, i.e. from a comprehensive trigram model the WER of its pruned version can be rapidly estimated. It is important to mention that this statement can be set out for these experiments thanks to the selected threshold since an improper beam setup might discard possible correct hypothesis.

This conclusion is valid for all languages included in this work. It can be seen in the case of English that the initial difference between trigram and trigram prune WERs is 1.5%, but as the system is more iteratively trained the difference between these two values becomes smaller. As a result, it can be said that the WER is going to be roughly  $\pm 1.5$  from the known value. The starting difference for German language is approximately 4.5% and similarly to English, this value decreases up to 2.5%. This behaviour can also be extended to Spanish language.

- The obtained WERs are strongly dependent on the set used to evaluate the recognition performance. Similar values of WERs were expected because the system used to recognise each set was the same depending of the language. Since no conclusion can be obtained from the values obtained at this stage of the methodology, it is decided to include both sets in the multilingual system. Nevertheless, it can be observed what set per language produces the better results. The lowest WER for English, Portuguese and Spanish are obtained by the evaluation set. For German the lowest WER is given by the development set.
- Portuguese exhibits the lowest accuracy. Despite the improvements of the WER through the different systems, the highest accuracy obtained for this language is not as good as the other languages. Because of this poor results it is expected to



**Figure 10** Comparison between the obtained WER when language model changes from trigram to trigram prune

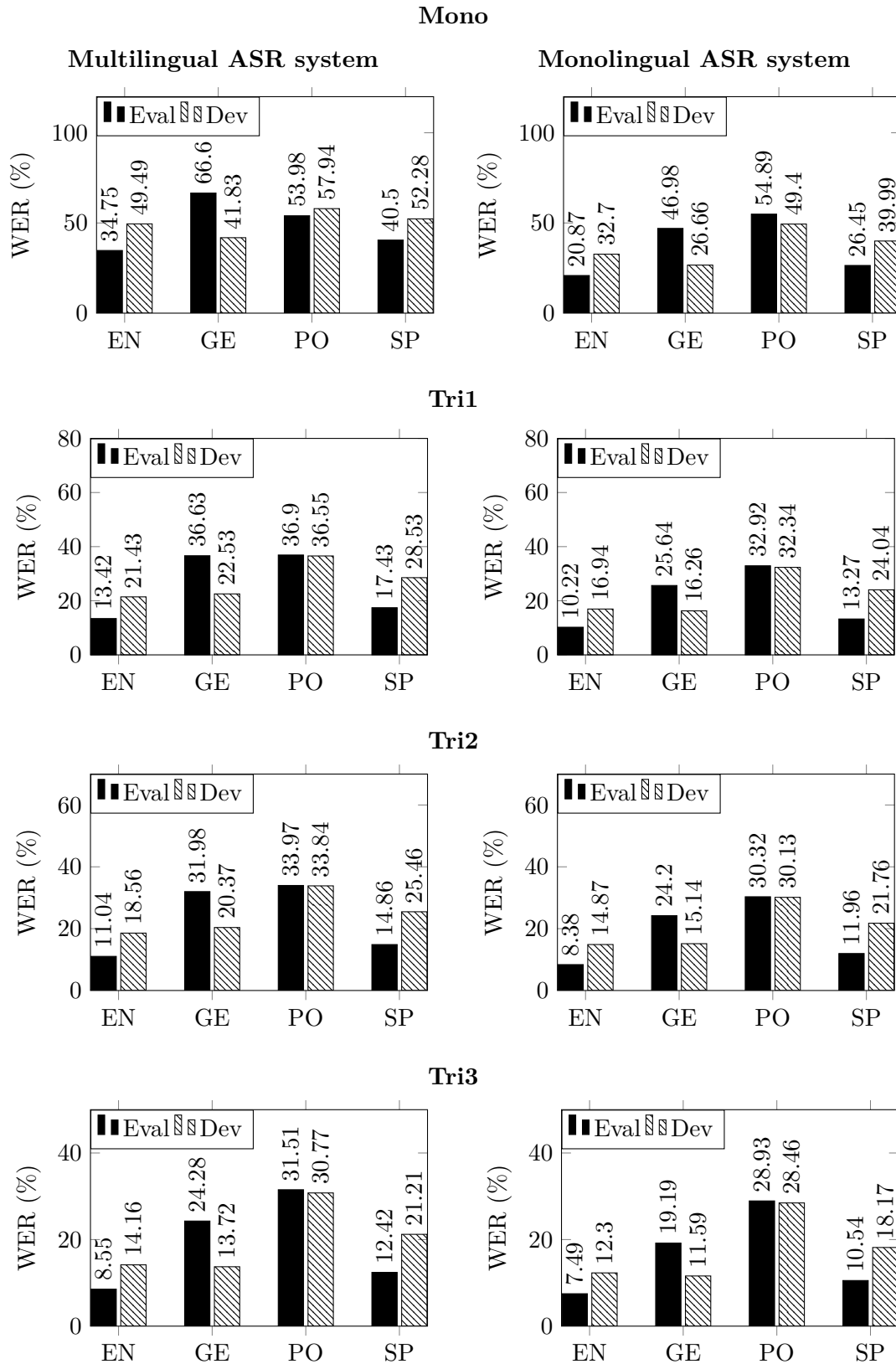
have a higher WER when this language is merged with the others. Because all GP data bases were automatically added in the recipe, a error in the code is discarded. Such poor error rate can be explain by the language model used (Sec. 3.4). The language model in this language presents the higher OOV which is around 10%.

#### 4.4.3 Shared acoustic modeling

Once the monolingual systems were successfully implemented, the integration of all languages into a shared acoustic model recognizer was followed according to the methodology. Table 21 presents the WER obtained for each language and Fig. 11 graphically compare the performance obtained with the shared acoustic model to the monolingual system results.

Some characteristics already seen in the monolingual system such as the dependence on the test set or the regular decrease of WER are still appreciated in the obtained results of the multilingual ASR system. Some interesting aspects are:

- The performance of the recognition task per language decreased. The reduction obtained are 1.44% for English, 3.61% for German, 2.45%, and lastly 2.46% for Spanish. This values were taken from the lowest WER results which were obtained by the system Tri3. Theoretically is known that when monolingual system are joint, the performance degradation is 5-9% compared to the monolingual systems. This range considers the fact when an acoustic model is shared [2]. Therefore, it can be said that the integration of the system was correctly implemented although some sets presented no poor results.
- The difference between the development an evaluation test of German language is not also presented in this study, but has also been matter of debate in different studies [43] and [44]. The difference results obtained by these two sets stems from the incomplete words that can be found in the German vocabulary. Thus any systematic error concerning the implementation of this multilingual system can be disregarded.
- English and Spanish are the languages with highest amount of training data since their number of non-shared phones is less than the number of Portuguese and German non-shared phones. This fact is reflected in the obtained WER for these two languages because they still are the languages with the lowest WER despite the integration
- The total CD phones without the phoneme set integration would have led to a total of 758 different phones. However, once the system applied the context dependency to the defined inventory list it could be seen that a reduction of 50.9% was obtained. Half of the phonetic units were reduced.



**Figure 11** WER results for the evaluation and development test when the sets are redefine to the size of the multilingual system

Set	Lang	Mono		Tri1		Tri2		Tri3	
		ML	Ref	ML	Ref	ML	Ref	ML	Ref
EN	Eval	34.75	20.87	13.42	10.22	11.04	8.38	8.55	7.49
	Dev	49.49	32.70	21.43	16.94	18.56	14.87	14.16	12.30
GE	Eval	66.60	46.98	36.63	25.64	31.98	24.20	24.28	19.19
	Dev	41.83	26.66	22.53	16.26	20.37	15.14	13.72	11.59
PO	Eval	53.98	54.89	36.90	32.92	33.97	30.32	31.51	28.93
	Dev	57.64	49.40	36.55	32.34	33.84	30.13	30.77	28.46
SP	Eval	40.50	26.45	17.43	13.27	14.86	11.96	12.42	10.54
	Dev	52.28	39.99	28.53	24.04	25.46	21.76	21.21	18.17

**Table 21** WER results for the evaluation and development test when the sets are redefine to the size of the multilingual system

## 5 Conclusions

A multilingual acoustic shared model speech recognition system was developed in this work. The developed ASR system was based defining a phoneme set that matches the similarities across the covered languages. Current system described in this work covers the following West-European languages: English, German, Portuguese and Spanish.

**Phonetic alphabet unification** X-SAMPA representation was used to unify the different phonetic representations used by covered languages in available resource, exactly it was CMUBET used in CMUDICT for English and alphabets used in GlobalPhone corpora for German, Portuguese and Spanish. Special attention has to be paid when the different representations are looked in the IPA representation since this is the bridge from any phonetic representation to X-SAMPA. Some changes had to be made in order to find a consistent representation across the languages. It was shown that the map was correctly resolved by different test. The preliminary test, using TIMIT, produced results to the ones that can be found in the literature. Additionally, the degradation of the WER when the shared acoustic model was implemented did not exceed the percentages that normally can be encountered when multilingual systems share acoustic parameters. Final combination of selected four languages reduced the size of the phoneme inventory by 50.9%.

**Kaldi implementation** The presented code was developed in such way that each new language can be implemented with rather small effort. In the case of GlobalPhone the integration could be done automatically. However, it always is important to check the output that the scripts used for the conversion of phones and change of encoding output the expected results. Different incoherent results were obtained during the development of this multilingual system since that information was not checked. If a manual mapping is desired to develop, the scripts containing the conversion of all languages can be reused.

**Final monolingual and multilingual systems** Created LVCSR system was tested using available corpora used also for AMs training and language models publicly available to anybody. The obtained WERs for monolingual systems for particular languages using two versions of language models (full-one and pruned-one) were very close, therefore the language model utilised in the multilingual system was just the pruned-one for each language. This decision also made the running times faster. Obtained WERs also varied according to the language. Previous studies have shown that results using the development set defined by the GP recipe for German produces better results. Although the results for Portuguese were the worst, the results of the evaluation and development sets remained almost the same with no prominent change depending on the set. There are different reasons that might be attributed to such result. Worse WER may be related to its language model since it has the highest rate of Out-of-Vocabulary words. The other one, concerning the regular WER between the sets, may be tied to the amount of data in these two sets. The difference between the words comprised by



the development and evaluation set of Portuguese is 2510. This difference is the lowest one in contrast to other languages; therefore it may explain the similarity of the results obtained for these two sets. English and Spanish always presented a rather low WER. Also, the error of these two languages did not significantly change from a monolingual system to the acoustic shared multilingual system.

**Futher work** This work has extended previous work on multilingual acoustic modeling developed for East-European languages and telephone speech [45]. It covers 4 West-European languages for which full-band acoustic data sampled by 16 kHz were available. The integration of both of these systems could be done to cover all available West- and East-European languages (in downsampled form of telephone speech). In addition to that, an improvement of the presented system could be achieved using further advanced acoustic modeling as SGMM, MMI or DNN-HMM based ones. Different ways to find each improve the recognition per language individually can be also explored. To use different languages models for Portuguese could decrease the obtained WER. In the case of German the results could be improved by trying another dictionary. Lastly, to avoid the different between the test sets, it should be suitable to collect more acoustic resources in order to count with more data in the development and evaluation sets, thereby the results by these two sets are similar.

## Bibliography

- [1] G. Saon, G. Kurata, T. Sercu, K. Audhkhasi, S. Thomas, D. Dimitriadis, X. Cui, B. Ramabhadran, M. Picheny, L.-L. Lim, *et al.*, “English conversational telephone speech recognition by humans and machines”, *CoRR*, 2017.
- [2] T. Schultz and K. Kirchhoff, *Multilingual speech processing*. Elsevier, 2006.
- [3] H.-P. Hutter, “Comparison of classic and hybrid hmm approaches to speech recognition over telephone lines”, PhD thesis, Swiss Federal Institute of Technology in Zurich, 1996.
- [4] J. Harrington and S. Cassidy, *Techniques in speech acoustics*. Springer Science & Business Media, 2012.
- [5] K. S. Rao and S. G. Koolagudi, *Robust emotion recognition using spectral and prosodic features*. Springer Science & Business Media, 2012.
- [6] M. Wand, “Advancing electromyographic continuous speech recognition: Signal preprocessing and modeling”, PhD thesis, The Karlsruhe Institute of Technology, 2015.
- [7] M. Sigmund, *Voice Recognition by Computer*. Tectum Verlag, 2003.
- [8] S. Young and G. Bloothoof, *Corpus-Based Methods in Language and Speech Processing*. Springer, 2013.
- [9] S. Sakti, K. Markov, S. Nakamura, and W. Minker, *Incorporating knowledge sources into statistical speech recognition*. Springer Science & Business Media, 2009.
- [10] Z.-H. Tan and B. Lindberg, *Automatic speech recognition on mobile devices and over communication networks*. Springer Science & Business Media, 2008.
- [11] H. Aghajan, J. C. Augusto, and R. L.-C. Delgado, *Human-centric interfaces for ambient intelligence*. Academic Press, 2009.
- [12] N. Jakovljević, M. Janev, D. Pekar, and D. Mišković, “Energy normalization in automatic speech recognition”, in *International Conference on Text, Speech and Dialogue*, Springer, 2008, pp. 341–347.
- [13] S. Greenberg, W. A. Ainsworth, A. N. Popper, and R. R. Fay, *Speech processing in the auditory system*. Springer, 2004.
- [14] Q. B. Nguyen, T. T. Vu, and C. M. Luong, “Improving acoustic model for vietnamese large vocabulary continuous speech recognition system using deep bottleneck features”, in *Knowledge and Systems Engineering*, Springer, 2015, pp. 49–60.
- [15] L. Rabiner and B.-H. Juang, *Fundamentals of speech recognition*. PTR Prentice Hall Englewood Cliffs, 1993.
- [16] D. Vasquez, R. Gruhn, and W. Minker, *Hierarchical neural network structures for phoneme recognition*. Springer Science & Business Media, 2012.

- [17] A. Waibel and K.-F. Lee, *Readings in speech recognition*. Morgan Kaufmann, 1990.
- [18] X. Zhang, “Rapid speaker and environment adaptation in automatic speech recognition”, PhD thesis, KU Leuven, 2014.
- [19] R. Rasipuram and M. Magimai-Doss, “Acoustic and lexical resource constrained asr using language-independent acoustic model and language-dependent probabilistic lexical model”, *Speech Communication*, vol. 68, pp. 23–40, 2015.
- [20] A. Khusainov, “Recent results in speech recognition for the tatar language”, in *International Conference on Text, Speech, and Dialogue*, Springer, Ed., 2017.
- [21] J. Benesty, M. M. Sondhi, and Y. Huang, *Springer Handbook of Speech Processing*. Springer, 2007.
- [22] W. de Gruyter, *Handbook of standards and resources for spoken language systems*. Gibbon, Dafydd, Moore, Roger, and Winski, Richard, 1997.
- [23] V. Romero, A. H. Toselli, and E. Vidal, *Multimodal interactive handwritten text transcription*. World Scientific, 2012.
- [24] G. Donaj and Z. Kačič, *Language modeling for automatic speech recognition of inflective languages: an applications-oriented approach using lexical dataage modeling for automatic speech recognition of inflective languages: an applications-oriented approach using lexical data*. Springer, 2016.
- [25] R. Rasipuram, “Grapheme-based automatic speech recognition using probabilistic lexical modeling”, *École polytechnique fédérale de Lausanne*, 2014.
- [26] M. Harju, P. Salmela, J. Leppänen, O. Viikki, and J. Saarinen, “Comparing parameter tying techniques for multilingual acoustic modelling”, in *Seventh European Conference on Speech Communication and Technology*, 2001.
- [27] D. B. Paul and J. M. Baker, “The design for the wall street journal-based csr corpus”, in *Proceedings of the workshop on Speech and Natural Language*, Association for Computational Linguistics, 1992, pp. 357–362.
- [28] T. Schultz, “Globalphone: A multilingual speech and text database developed at karlsruhe university”, in *Seventh International Conference on Spoken Language Processing*, 2002.
- [29] I. P. Association, *Handbook of the International Phonetic Association: A guide to the use of the International Phonetic Alphabet*. Cambridge University Press, 1999.
- [30] D. Gibbon, I. Mertins, and R. K. Moore, *Handbook of multimodal and spoken dialogue systems: Resources, terminology and product evaluation*. Springer Science & Business Media, 2012.
- [31] J. Durand, U. Gut, and G. Kristoffersen, *The Oxford handbook of corpus phonology*. Oxford Handbooks in Linguistic, 2014.
- [32] J. Klautau Aldebaro Barreto da Rocha, “Speech recognition using discriminative classifiers”, *University of California, San Diego*, 2008.
- [33] *Globalphone: Dictionaries in multiple languages - german -*, XLingual, GmbH Co. KG, Dec. 2012.
- [34] *Globalphone: Dictionaries in multiple languages - portuguese -*, XLingual, GmbH Co. KG, Jan. 2014.

## Bibliography

- [35] *Globalphone: Dictionaries in multiple languages - spanish -*, XLingual, GmbH Co. KG, Mar. 2013.
- [36] 2012. [Online]. Available: <https://www.uni-bremen.de/en/csl/projects/current-projects/global-phone-language-models.html%5C>.
- [37] T. Schultz and T. Schlippe, “Globalphone: Pronunciation dictionaries in 20 languages”, in *LREC*, 2014, pp. 337–341.
- [38] K. T. Riedhammer, “Interactive approaches to video lecture assessment”, PhD thesis, University of Erlangen-Nuremberg, 2012.
- [39] K.-F. Lee, *Automatic speech recognition: the development of the SPHINX system*. Springer Science & Business Media, 1988.
- [40] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hanemann, P. Motlicek, Y. Qian, P. Schwarz, *et al.*, “The kaldi speech recognition toolkit”, in *IEEE 2011 workshop on automatic speech recognition and understanding*, IEEE Signal Processing Society, 2011.
- [41] Jan. 2016. [Online]. Available: <http://jrmeyer.github.io/asr/2016/02/01/Kaldi-notes.html>.
- [42] P. J. Moreno, “Speech recognition in telephone environments”, Master’s thesis, Carnegie Mellon University, 1991.
- [43] M. Erhardt, D. Telaar, and T. Schultz, “Error blaming based on decoding output”, 2013.
- [44] A. Mohan and R. Rose, “Multi-lingual speech recognition with low-rank multi-task deep neural networks”, in *Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2015, pp. 4994–4998.
- [45] J. Fiala, “Dnn-hmm based multilingual recognizer of telephone speech”, Master’s thesis, Czech Technical University in Prague, 2016.