Universität Zürich[UZH]

Master thesis
zur Erlangung des akademischen Grades
Master of Arts
der Philosophischen Fakultät der Universität Zürich

(Titel)

Author: Deborah N. Jakobi
Matriculation number: 16-054-165

Supervisor:

Institut für Computerlinguistik

Abgabedatum: (xx.xx.xxxx)

Abstract

This is the place to put the English version of the abstract.

Zusammenfassung

Und hier sollte die Zusammenfassung auf Deutsch erscheinen.

Acknowledgement

I want to thank X, Y and Z for their precious help. And many thanks to whoever for proofreading the present text.

Phillip Ströbel from the CL institute at the UZH for his help with the OCR technologies. Lysander Jakobi for writing the Hebrew transcription. Florina Vogel for helping with the Farsi transcription. Si-En? Tanzil?

Contents

List of Figures

List of Tables

List of Acronyms

IPA International Phonetic Association

IPA International Phonetic Alphabet

WER word error rate

CER character error rate

G2P grapheme-to-phoneme

seq2seq sequence-to-sequence

SIGMORPHON Special Interest Group on Computational Morphology and Phonology

NLP natural language processing

PoS Part-Of-Speech

WALS World Atlas of Language Structures

ASR automatic speech recognition

TTS text-to-speech

HTR Handwritten Text Recognition

DISC distinct single characters

FST finite-state transducers

EM Expectation-Maximization

1 Introduction

With the advent of technologies that can process huge amounts of data, many linguistic tasks that were originally very tiresome and expensive to do, can now be accomplished much faster. Well known examples for this branch called natural language processing (NLP) are machine translation or search engines. A lot of available tools and consequently research done in this area is concerned with written text. For many scenarios like machine translation it is well possible to argue that this makes a lot of sense. Following, there is an ever-growing set of models that are trained on text and are used to accomplish those text-based tasks. Often the goal is to reach or outperform human solutions to those various tasks. From a linguistic point of view, the questions comes up if focusing on written language only can ever represent human language adequately. Most of communication and daily language use happens through speaking. This is a first potential limitation to many (written-)language technologies. Another possible limitation is the concern if written text represents language in general well enough to draw significant conclusions. It is not easy to find out what characteristics of a language can be observed in written representations of a language. There are technologies like automatic speech recognition (ASR) or text-to-speech (TTS) that require the mapping of written to spoken language. Spoken language in those cases is mostly represented as phonetic transcriptions as those are easier to process. They do contribute to the questions how spoken and written language relate. But this still does not answer the question of how well written language represents language in general. The representative power of written text is much less studied. This is where this current thesis connects to cutting-edge research. I am going present my attempt of studying a multilingual phonetic corpus and comparing it to its written-text version. I will try to answer the following final question: **Is it essential for the study of multilingual corpora to perform analyses on phonetic text (i.e. speech representations) rather than only written text?** It becomes clear, when looking at these considerations, that there are a few huge topics addressed. None of these is trivial and can be answered easily. While this thesis cannot possibly discuss everything from the use of phonetic transcriptions up to the nature of human language use, the aim is to make a step into the direction of quantifying the representative power of written text.

1.1 Research questions & goals

The text group of the Language and Space lab at the University of Zurich maintains a project that provides a multilingual corpus consisting of 100 language text samples [SPUR project]. Those 100 languages are meant to be representative for all the world's languages which is explained in more detail in section 2.4. It is therefore meant to give insight on relations, similarities, differences or properties of individual languages or language families. Specifically, their goal is to use quantitative methods like statistical modelling, machine learning and information theory to study language variation and compare languages. The goal is now to collect phonetic transcriptions of the corpus. The same analyses that are performed on the original written corpus can be performed on the phonetic texts and both can be compared. In order to add a phonetic corpus to the already existing one, various steps need to be performed which are outlined below:

1. Data collection: The given dataset contains no phonetic transcriptions of those 100 languages. The first step is to find already existing data.
2. Phonetic transcriptions: As existing data will not be available in sufficient amounts to perform meaningful analysis, the next step is to actually create phonetic transcriptions of as many languages as possible of the corpus.
3. Calculations and Analysis: Once the transcriptions have been obtained, the newly created phonetic corpus can be analysed and calculations can be performed.

By performing these steps I am aiming at answering the following two questions:

1. Is there any significant difference in comparing spoken or written languages?
2. Does written text represent language well enough to justify text-based research only?

1.2 Thesis structure

The thesis is subdivided into six chapters including a final conclusion. Chapter 3 sets the boundaries of the theoretical background. It presents the linguistic foundation of phonetics and phonology, an introduction to corpus linguistics or rather corpus phonetics and finally an overview of the possibilities for automated creation of phonetic transcriptions. Chapter 4 introduces to the struggle of data collection.

It explains the various data types and how those can be used. Chapter **??** dives deeper into the possibilities for creating phonetic transcriptions and what models can be used to create those. Chapter 5 presents my own experiments to create phonetic transcriptions of the corpus.

2 Linguistic Background

2.1 The relation of spoken and written language

corpus linguistics and quantitative analysis. Remember that writing systems came only much later compared to language in general. Can they capture language as such well enough? Computational linguistics deals mostly with written languages, what does linguistics say and do?

Whenever we study language we look at samples of that language. It is simply impossible to study an *entire* language as we would need all texts that were ever produced in that language. Consequently, we need to ask ourselves how much material of a language is enough to study it properly [Baird et al., 2021]. In addition, language material can represent written or spoken language. We will see later on in this chapter that mapping a spoken language to its written representation is far from easy and never perfect. Baird et al. [2021] focus on answering the question how much phonetic data is needed to represent a language well.

2.2 Phonemes and syllables

Given that phonetics and phonology is a sub-area of traditional linguistics and often only touched on superficially in computational linguistics, I will summarise the most important assumptions and terms concerning said field. A very important terminological distinction is between phonetics and phonology. While phonetics refers to the study of actual sounds, phonology refers to the study of sound *systems*. In phonetics, it is not so much important what the different sounds mean, but how they are produced and perceived and what different sounds a human being can produce and perceive at all. When it comes to human communication using spoken language, many of these sounds are not actually used to produce distinguishable meaning. This is why on the other hand phonology is important to describe the set of distinguishable sounds that make up a language. For example: the letter /r/ in English can be pronounced in many different ways. None of those pronunciations

produces a change in meaning. This means that there exist many different *phonetic* sounds but only one *phonological* or *phonemic*. Those sounds are referred to as phone and phoneme respectively. While there are infinitely many phones there are only finitely many phonemes in a language. Sounds that can be interchanged with another sound without changing the meaning are referred to as 'allophones'. Not all different possible sounds are actually considered qualitatively 'good' sounds of a language. Usually there is a subset of all possible phones that is accepted as 'good quality sounds' within all different dialects of a language [Kracht, 2007]. An obvious example being loudness: Although very silent speech produces correct phones, these are not 'good quality' as they simply cannot be understood. Or speaking in English with hardly any mouth and tongue movement. Although this produces understandable sound, it is not generally considered good speech.

It is important to note at this point that the terms phonetic and phonemic respectively phone and phoneme are sometimes used interchangeably. Their linguistic definition as given above is clear while the definition on the computational side is often less strict. Strictly speaking phonemic transcriptions are not allowed to contain allophones but should write the respective phoneme. This will not always be the case when it comes to data used in language technology [Lee et al., 2020].

Vowels and consonants Each phoneme can be described based on different categories. A well-known distinction is that between vowels and consonants. Both of these are again categorized differently. The schema for vowels and consonants is inspired by the human vocal cavity. The terms to describe vowels sounds are based on the position of the tongue in the mouth and if the lips are rounded or not. Using those two categories enables us to distinguish every possible vowel. Consonants are defined by the place and the manner of their production. The place, again, refers to the position of the tongue in the mouth and the overall form of the vocal tract. The vocal tract is used to block the air and make it flow in a specific way. The manner, on the other hand, describes the way the air is lead through the mouth or how it is blocked to produce a sound [CrashCourse, 2021a]. finish consonant part. The exact description of each phoneme will later become important when we talk about representing phonemes for a grapheme-to-phoneme (G2P) model in section 3.6. Check this again if this is needed. add explanation of categorization of consonants

Syllables Phonemes, or letters, can be grouped into larger units called *syllables*. Syllables can be an entire word or a part of a word. English syllables typically consist

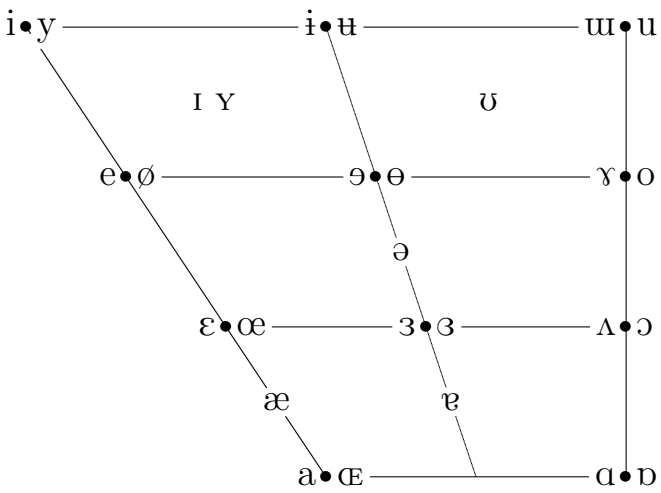

Figure 1: The figure represents the vowel diagram as presented by the IPA. The chart is meant to represent the vocal cavity of a human being from the side, with the mouth opening to the left. The edges and consequently the vowels are named analogous to the position of the tongue in the vocal cavity. The upper left vowel is called close front vowel. The upper right vowel is called close back vowel. The two vertical lines in the middle are half-close respectively half-open. The middle vertical line is simply called mid. I am not sure if I should keep it, if I do, I will add a consonant chart as well. MAybe I'll put it into the appendix

of a group of consonants followed by a group of vowels or a diphthong followed by a group of consonants again. These parts are called *onset*, *nuleus* and *coda* respectively. For every syllable in every language it is true that the nucleus cannot be empty. The onset and the coda can be empty. Other than that, syllables are organized very differently in different languages. [Kracht, 2007]

Tones add explanation of monophthongs, diphthong, suprasegmental they appear quite often in the lit (maybe make a glossary

2.3 Mappings of written and spoken language

Unlike spoken language that was a part of human interaction all the time, writing systems only developed over time. There are different writing systems that developed in different places at different times. The structure of the spoken language, the cultural context or the tools that were at hand to write are a few of many factors

that influenced the emergence of a specific writing system. In General, we can think of writing systems as mappings from spoken language to written language. The systems used to represent sounds in different languages do not uniquely map a letter to one specific phoneme. Most of the time, there is a standard pronunciation of each letter that is trained by reciting the alphabet. However, in reciting the alphabet there is a vowel added to the consonants in order to pronounce them more easily. These explanations make clear that the mapping of written text to spoken text in various languages is complex. When taking a step back, we can see that a single grapheme can represent either a phoneme, a syllables or words. Each mapping will be presented below:

Alphabet When a grapheme maps to a phoneme, we call this an alphabet. In German, for example, the writing system consists of the Latin alphabet. The Latin alphabet is used for many different languages in western Europe and those languages that were influence by colonisation. There are other alphabets like the Cyrillic or the Greek alphabet. Having an alphabet does not mean that each grapheme, or letter in this case, maps to exactly one phoneme. In fact, one grapheme can have many different realizations as example 2.1 shows.

(2.1) The examples show the different realizations of the English grapheme sequence 'ough' [CrashCourse, 2021a]

(a) tough [tʌf]

(b) cough [kɒf]

(c) though [ðəʊ]

(d) through [θruː]

(e) bough [baʊ]

(f) brought [brɔːt]

The above examples show that it is not possible to have a one-to-one mapping from one grapheme or a sequence of graphemes to one phoneme or a sequence of phonemes with in the English language. Let alone within all languages that use the Latin alphabet. In addition, alphabets typically have diacritic marks that can be used to extend the main letters. Just as with single graphemes, also diacritic marks cannot simply be mapped to a phoneme.

Abjad A special variant of an alphabet-language is abjad. Abjad represents only consonants and no vocals. Semitic languages like Hebrew or Arabic make use of abjad.

(2.2) Hebrew examples that are first mapped to Latin alphabet then to phonemes.

(a)

SYLLABARY In syllabaries, a grapheme represents a syllable instead of a single sound. Examples are the Japanese Hiragana and Katakana.

LOGOGRAPHIC SYSTEMS Logographic systems represent entire words or morphemes as graphemes. Chinese is an example for a logographic system. We cannot break down Chinese signs into single morphemes or letters.

What all of these mappings have in common is that they are no reliable source of pronunciation [Kracht, 2007].

The history and development of writing systems is an entire independent study area. For this thesis it is mostly important to be aware of the independently developing systems. Not all scripts can be treated the same and this most certainly has implications on models to create phonetic transcription.

An exception to the above explained characteristics of an alphabet are phonetic alphabets like the International Phonetic Alphabet (IPA) where each grapheme represents exactly one phone [CrashCourse, 2021b; Kracht, 2007]. More on this special alphabet will be explained in section 4.1.

Many of the pronunciation rules of a language are based on convention. Speakers of a language just *know* how to pronounce a word. Still, there can arise heated debates about the correct pronunciation of certain words. add some info here. Apart from these conventions, spoken and written languages change differently over time. Spoken languages are typically more flexible and ready to change while their written representation often stays the same [Moran and Cysouw, 2018]. This can lead to official governmental interventions like the German orthography reform of 1996 that intended to adapt the German spelling to represent the German pronunciation more adequately. Also, major inventions like printing machines gave rise to standardization of writing systems as reading and writing became more common.

2.4 The corpus

As mentioned int eh introduction, the basis of the data used in this thesis is a corpus provided by the SPUR lab at UZH. The corpus contains 100 languages which are proposed by Comrie et al. [2013]. This online book contains different chapters

each of which shows a different linguistic feature including a map which shows the distribution of that feature over the world's languages. While the number of languages presented on the individual maps depends on the amount of research done in a specific area, the sum of all maps gives quite an impressive overview on the structure of nearly half of the world's languages. Out of the 2676 languages a sample of 100 languages was chosen. This sample does not contain too many languages from one area, neither does it contain too many languages from one family. Not considering the aforementioned criteria of maximizing genealogical and areal diversity can lead to misleading results. Figure 2 shows the distribution of the corpus on a world map. The different icons show the genus of the languages which is a classification of languages defined by the World Atlas of Language Structures (WALS) team that maintains the language collection. The interactive map can be viewed online [100-language-sample]. Table **??** in the appendix A shows all languages that are in the 100 language corpus. None of the text samples are provided by WALS. The entire corpus is provided by the SPUR team that collected the corpus over the last few years and is continuously working on and with it.

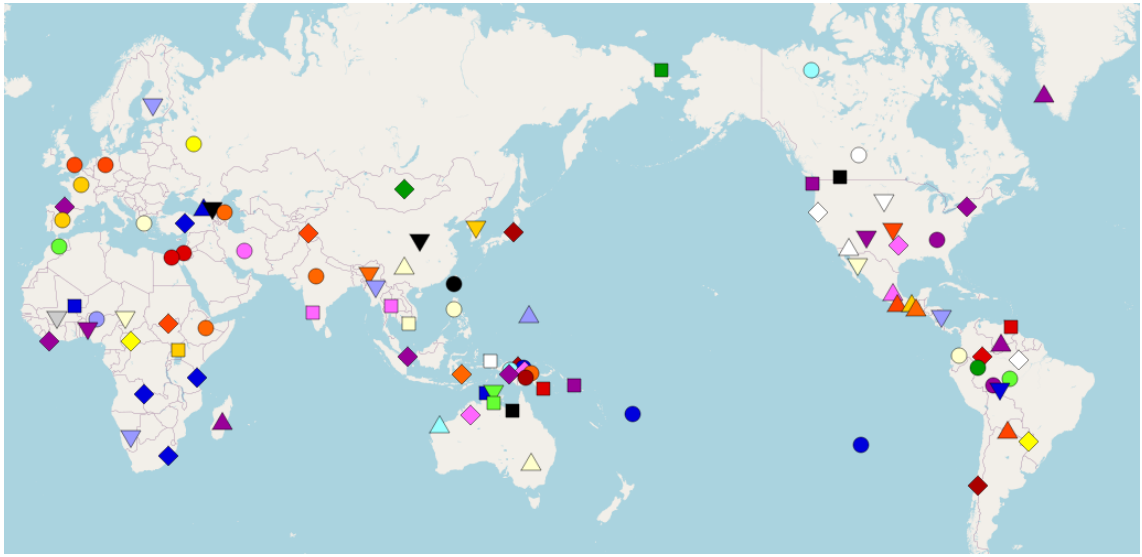

Figure 2: WALS - 100 Language Sample

2.5 Corpus phonetics

Due to recent technological advancement it has become possible to store large digital collections of speech recordings and their aligned transcriptions. These possibilities gave rise to a wider acknowledgement of corpus phonetics. Corpus phonetics deals with an abundance of linguistic variation. In addition to language, style or vocabu-

lary variation, there are differences in dialect and idiolect, physiological state of the speakers and their attitude [Liberman, 2019; Chodroff, 2019]. Many methods and tools used in corpus phonetics are based on ASR algorithms or simple programming [Chodroff, 2019].

A way to analyse or use phonetic corpora is to use phonetic features to represent each phoneme. These features are a list of properties that are overlapping with the phonetic description of each phoneme. It is a list that can minimally be used to describe the phonemes.

3 Technical Background

This chapter presents the technical background that is needed for this thesis. I will first present general architectures and frameworks that are commonly used and then present current models used for creating phonetic transcriptions. Table 1 shows the state-of-the-art models for G2P modelling.

3.1 Automated phonetic transcription

Today's technologies allow to build models that create phonetic transcriptions automatically given an original text. There are several approaches which will be discussed below. Creating phonetic transcriptions is essentially a seq2seq task. Like other NLP tasks its goal is to transform a sequence of characters into another sequence of characters. In the present case, the input sequence is a sequence of graphemes. These can look very differently depending on the script (see section 2.3). The output sequence is a sequence of phonemes[1]. A common way to transform written text into its phonetic version is referred to as G2P. The idea behind this approach is that individual letters (graphemes) are converted into sounds represented as phonemes.

Most of the research done in this area is limited to the English language. This is not uncommon in NLP research. The overwhelming availability of English data resources and the unavailability and serious struggles to find data in other languages heavily influences this research. Ashby et al. [2021] report that the SIGMORPHON (Special Interest Group on Computational Morphology and Phonology) G2P tasks in 2020 and 2021 is the first attempt to tackle multilingual G2P. While these are very recent tasks, there are earlier models and methods that contributed to the evolving of nowadays G2P methods.

[1]Please refer to section 2.2 in order to understand the terminological implications of phoneme. As it is common in research, I will stick to the term *phoneme* although strictly speaking it is not always correct. Phoneme in this case just refers to any symbol that is used to represent a sound.

Author	Model Architecture	ISO 639-3	WER
SIG21: Clematide and Makarov [2021] Link	CLUZH models 1-7. LSTM-based neural transducer with pointer network-like monotonic hard attention trained with imitation learning. All models 1-7 are majority-vote ensembles with different number of models (5-30) and different inputs (characters or segments). Achieved good results in nld (14.7), ice (10), jpn (5.0), fra (7.5) and vie (2.0) but not better than SIG20.	medium (10,000 pairs)	
		hye (arm_e)	6.4
		hun	1.0
		kat (geo)	0.0
		kor	16.2
		low (800 train pairs)	
		ell (gre)	20
		ady	22
		lav	49
		mlt_ltn	12
		cym (wel_sw)	10
SIG21: Lo and Nicolai [2021] Link	UBC-2 outperforms the baseline. They analysed the errors of the baseline and extend it by adding penalties for wrong vowels and wrong diacritics. Errors on vowels actually decreased. Best macro average (low -resource).	ady	22
		khm	28
		lav	49
		slv	47
SIG21: Gautam et al. [2021] Link	Dialpad-1: Majority-vote ensemble consisting of three different public models (weighted FST, joint-sequence model trained with EM and a neural seq2seq), two seq2seq variants (LSTM and transformer) and two baseline variations.	high (32.800 train pairs)	
		eng (eng_us)	37.43
SIG20: Peters and Martins [2020] Link	DeepSPIN-2,-3,-4: Transformer- or LSTM-based enc-dec seq2seq models with sparse attention. Add language embedding to enc or dec states instead of language token.	3.600 train pairs	
		jpn (jpn_hira)	4.89
		fra (fre)	5.11
		rum	9.78
		vie	0.89
SIG20: Yu et al. [2020] Link	IMS: Self training ensemble of one n-gram-based FST and 3 seq2seq (vanilla with attention, hard monotonic attention with pointer, hybrid of hard monotonic attention and tagging model).	hin	5.11
		nld (dut)	13.56

Table 1: This table presents the state-of-the-art G2P models. Models that are important for this thesis will be explained in more detail. The language code in parenthesis is the code used in the respective paper.

3.2 Evaluation metrics

The most common metric to evaluate phonetic transcriptions is the word error rate (WER). This is the percentage of predicted transcriptions that deviate from the gold standard. The lower, the better the model. The idea is that we can capture

the cost that it takes to transform the system text into the reference text. If the WER is 0, this means that the texts are exactly the same. The following formula is used to calculate WER:

$$WER = \frac{S + I + D}{N} \tag{3.1}$$

In equation **??** the S stands for substitution, I for insertion, D for deletion and N denotes the total number of words in the reference sequence. If you want the percentage the number needs to be multiplied with 100. Note that the WER can be more than 100%. This happens if, for example, the are a lot of additional insertions or deletions in the system text. Another metric that is used quite often is the character error rate (CER). It is caculated in the exact same way as the WER, but instead of words everything is calculated on character basis. In a multilingual setting, it is sometimes necessary to have a score for the entire system covering more than one language. In such cases it is custom to use a macro-averaged WER or CER. explain macro average (and micro to be complete)

quote Blog link

3.2.1 Rule-based models

The first systems to create phonetic transcriptions of text were rule-based systems. Rule-based transcriptions models are built using linguistic pronunciation rules. In order to be able to create such a system, one needs to collect pronunciation rules first. While there are only few languages where such rules are ready and available for the general public there are many languages where those rules need to be created first. In order to create the rules in the first place, a lot of linguistic expertise is needed. Apart from this initial effort to create the rules, a problem with rule-based approaches is the maintenance of the systems. To maintain the system, experts need to keep track of language change which is time consuming and expensive. In addition, most languages are irregular in their pronunciation and those irregularities need to be tracked. Due to the open-vocabulary situation and the impossibility to cover all possible words, all systems must be able to deal with rare and unseen words [Rao et al., 2015; Bisani and Ney, 2008]. Rule-based systems are outperformed by more recent neural systems [Gorman et al., 2020; Ashby et al., 2021]. Many earlier systems published considered only one language and were not multilingual (see e.g. Toma and Munteanu [2009]).

Epitran However, there are languages that are more or less regularly pronounced. The Epitran system makes use of this and presents a rule-based system for G2P conversion for mostly low-resource languages. The system has the ability to provide a solution for every possible word and is consistent within its transcriptions. Epitran for all languages except English and traditional Chinese works with a map file that allows to map graphemes to phonemes. Additional pre- or post-processing can be applied that follow context specific rules [Mortensen et al., 2018].

[Rao et al., 2015]

[Bisani and Ney, 2008]

add a few examples of rule-based systems and why and by whom they where outperformed (see Ashby et al. [2021]; Gorman et al. [2020] for this purpose)

3.2.2 N-gram Models / Statistical models

N-gram models, statistical models or joint-sequence models were used before neural models took over the field. These are sometimes referred to as traditional models. One reason why they were outperformed by neural models is that it is necessary to construct alignments between grapheme and phonemes. This is necessary because one grapheme can be realized as multiple phonemes or vice versa. It is not possible to simply have a one-to-one alignment. Joint-sequence models were often used with different versions of the EM algorithm. Other statistical models include weighted FSTs [Lo and Nicolai, 2021].

check Lo and Nicolai [2021] they include a lot of references about this topic

transducers: those are like automaton. Unlike automaton that only tell you if a certain sequence is in a particular language, transducers output something at every state.

3.2.3 Neural models

Neural G2P models have been reported to outperform most other models [Lee et al., 2020]. Many researchers experiment with different variants of LSTM models [Lee et al., 2020; Hammond, 2021; Gautam et al., 2021; Rao et al., 2015]. But there are also other models that have been used. All of those will be introduced in the following.

LSTM LSTM means long short-term memory. They are inspired by the simpler RNNs. As their name suggests they include what we could call a memory. Instead of just updating the state with all the at every step through the sequence, LSTMs can more flexibly decide what information is added to the state and what information should be forgotten. Many earlier neural LSTM models use a connectionist temporal classification layer to include alignment information [Lo and Nicolai, 2021].

Transformer

Neural Transducer Neural transducers, as presented by Jaitly et al. [2016], extend previously used seq2seq models. They can treat more arriving input without having to redo the entire calculation for the entire updated sequence. At each time step, the neural transducer can output zero to many output symbols.

seq2seq: condition output sequence on entire input sequence. This does not work well for input that gets continuously longer or very long input sequences.

A problem with creating phonetic transcriptions is that the input and output segments are not always of the same length. It is difficult to align input and output.

Generally, there is a difference between models that assume conditional independence between the each output step (e.g. Hidden Markov Models) and there are models that do not make this assumption but condition the current output on the entire sequence before (seq2seq). Seq2seq models, however, have to wait until the full input sequence is processed before they can start decoding.

3.2.4 SIGMORPHON shared tasks

The Special Interest Group on Computational Morphology and Phonology (SIGMORPHON) [Sigmorphon, 2021] regularly organizes shared tasks concerned with morphology and phonology. For the years 2020 and 2021 they organized a G2P conversion task [Ashby et al., 2021; Gorman et al., 2020]. The tasks represent a first attempt at creating benchmarks for multilingual G2P conversion. Both tasks and their results will be discussed in sections 3.2.4 and 3.2.4. Although there is other research on G2P, many recent publications have been made within the SIGMORPHON shared tasks which is why there are two separate sections on those tasks. As the SIGMORPHON tasks are the most recent and probably most influential contributions to G2P research, both tasks will be discussed separately below.

Yu et al. [2020] contributed to the 2020 SIGMORPHON G2P task. Their contribution is of particular interest for this thesis as it proposes a data augmentation model for low-resource settings. As there are many languages in the corpus that have only very little available data, such a model could be of great use. The methodology applied in their approach is ensemble learning combined with a self-learning strategy. finish summary on 2020 sig task

The second iteration of this G2P task attempts at outperforming the models of the previous task. An additional challenge is its separation into high-, medium- and low-resource languages. This reflects the needs of this present research well, as many languages in the corpus are low resource languages. In preparation for the task, the WikiPron data (see chapter 4) was cleaned to exclude foreign words that include phones that are not in the actual language's native phone inventory. If a word contained foreign phones, it was excluded. This was the case for words whose pronunciation was not adapted to the language at hand but the transcription of the foreign language was used. This cleaning was only applied to medium- and low-resource languages. Additionally, the lists were sorted according to scripts. There are many languages that use multiple scripts and using them in the same dataset does not produce good results. There were other steps done to ensure good quality of the datasets. I collected more on that subject in section 3.4. The final datasets have to following sizes: The high-resource subtask consisted of about 41,000 word-transcription pairs of American English only. The medium-resource task provided 10,000 word-transcription pairs for ten languages and the low-resource task another 1,000 for ten different languages [Ashby et al., 2021]. The datasets were split into 80% training data, 10% development data and another 10% test data.

The baseline for this year's G2P task is an adapted version of last year's submission by Makarov and Clematide [2020]. The baseline model has been made available for this year's task. The model they use is a neural transducer that is trained with imitation learning. The basis of the neural transducer was originally designed for morphological inflection [Aharoni and Goldberg, 2016]. Instead of just learning to output the correct string, the model learns to produce an optimal sequence of edit actions needed to transform the input string into the output string. Due to the nature of inflection (overlapping alphabets of input and output sequences), the original model was encouraged to copy the input. This does not work well for G2P tasks as the input and the output alphabet are not always the same (especially for non-Latin scripts like Korean). explain neural transducer, the model more in depth.

Lo and Nicolai [2021] chose to perform an error analysis on the baseline and try to minimize the frequent errors for the low-resource setting only. The analysis showed

that often the model gets vowels and diacritics wrong. The extended the baseline in a way such that wrong vowel and diacritics predictions are punished more than other errors. This model outperformed the baseline for some languages (ice, khm, lav, mlt, slv) and was level for another (ady). The predictions with their model, shows an improvement in vowel prediction. A further analysis showed that many errors still happen with vowels. Vowels get often confused with similar vowels. Their conclusion is that many of these errors make sense in a linguistic sense. They also tested augmenting the input data with syllable boundaries and using the baseline model as-is which did not improve the results.

As explained above, the model learns to create sequences of edit actions. The problem with this approach is that there are many possible sequences of edit actions that produce the same result. Imitation learning is proposed as a solution for this problem. explain imitation learning better and more precise.

Results add 2020 results The results of the 2021 task show that there are great differences in languages. One possible explanation is that the datasets were a mix between broad and narrow transcriptions. As narrow transcriptions contain much more detail, it can be argued that this is more difficult for any system. The authors doubt the influence of this but they could not (yet) quantify this impression.

Results in the low-resource setting are still worse compared to the medium-resource setting. This means that current systems seem to be unable to achieve a good performance when only using 800 samples for training. More research needs to be done in data augmentation techniques and improving the systems to cope with only little available data.

The differing performance for various languages calls for the questions what makes a language hard to pronounce. Especially as for Georgian which was in the medium-resource setting all submissions and the baseline reached a WER of 0.0. Interestingly enough, the WER for the language in the high-resource setting, English, reached one of the highest WERs.

Error analysis add 2020 error analysis In order to find the most common errors of the systems, there were two types of analysis conducted. The first one is to simply find the most common wrongly transcribed grapheme-phoneme pairs. This analysis showed that many errors are due to language internal ambiguities. Some errors go back to errors or inconsistency in the data. The other type of analysis is to create a covering grammar. This means that a grammar is created that includes all

possible combinations of grapheme-to-phoneme mappings that are allowed in this language. This set is constructed manually. This error analysis was only conducted for three languages in the medium setting and four of the low-resource languages. Then, words were considered that were predicted wrongly by the system. For those words it was checked whether the prediction by the system was completely wrong or if it was one of many possible transcriptions of that word. If so, the error was that system did not guess the correct transcription for this word. These errors could be considered ambiguities in the language. Another error type that can be identified by this analysis is when the reference transcription cannot be derived from the covering grammar. This can either mean that the covering grammar is incomplete or that the reference is a word that is very atypical for the language (e.g. borrowed word but not yet adapted to pronunciation rules) or simply wrong.

3.3 CMUSphinx

This model has been used on the SIGMORPHON task. It was not used on many language but promised a good performance which is why I decided to use this model for this present thesis. add explanation of model

3.4 Data quality considerations

not sure where to put this, but I think it makes sense to have a separate section on data quality. Most papers include some things Data quality is crucial in any machine learning application. Authors mostly include a section about their preprocessing and what should be done to ensure high quality datasets. The list given below is an incomplete list of potential problems and measures taken in different settings for G2P data:

- **Exclusion of words with less than two Unicode characters or less than two phone segments** [Ashby et al., 2021] add an explanation
- **Separation by script** [Ashby et al., 2021]: It is very straightforward why this is done. There is no obvious connection between the different scripts of a language and its pronunciation. It makes sense to treat different scripts as different languages.
- **Exclude foreign words with foreign pronunciations** [Ashby et al., 2021]: Foreign words in a language with their original pronunciation can add phonemes

that are not in that language's phoneme inventory. If they were to be included it would make sense to include a pronunciation adapted to the actual language.

- **Words with multiple pronunciations in word lists**: Ashby et al. [2021] excluded those words, however, it might also be possible to add Part-Of-Speech (PoS) tags or other linguistic information to distinguish these words.
- **Consistent broad transcriptions** [Ashby et al., 2021]: With broad transcriptions it is important to be consistent and not use allophones. Ashby et al. [2021] did this specifically for Bulgarian.
- **Linguistic variation and processes** [Ashby et al., 2021]: Some transcriptions include examples for monophthongization or deletion which are ongoing linguistic processes but should not be part of a dataset representing a standard variation. Ashby et al. [2021] dealt with monophthongization by choosing the longer to two transcriptions as this logically exclude the monophthonged version. This does of course only work if there are more than one pronunciations available.
- **Tie bars**: Ashby et al. [2021] notice that some languages (English and Bulgarian) have inconsistent use of tie bars. This can be correct by replacing all inconsistencies by the tie-bar-version.
- **Errors in the transcriptions**: Gautam et al. [2021] noticed many errors in the WikiPron English data. They identified errors by looking at the least frequent phones and then check the word-pronunciation pairs where those phones occurred in. As the number of phones in a language is often known this can be used to check the phones in the datasets and identify uncommon ones.

Especially the task of finding errors in the transcriptions is quite tricky. It requires a lot of knowledge about the phonology and phonetics of a specific language.

3.5 Unicode

When it comes to representing characters in a machine-readable format things get very tricky, very quickly. In order to understand this fundamental problem it is necessary to understand the basic concept behind unicode and encodings in general. As discussed in chapter 2, there are many different kinds of what we typically call letters or graphemes or characters. Just as a human writer must be able to uniquely identify each different letter or sign, so must a computer. The most widely spread

standard to represent scripts is called Unicode. Letters are mapped to unique numbers that can be rendered differently depending on the font and the context. There are different stages of representation until a letter can be represented on screen:

Code point A unique numerical, non-negative value usually expressed as a hexadecimal number (U+0000). Allows one-to-one mapping between letters and codes. Each code point has a set of properties attributed to it. Properties like the script, uppercase or not, etc.

Character An abstract representation of the shape of the grapheme. Can in theory not be represented visually, as this includes a font. A Unicode character is *not* the same as what we would call a letter or a sign in different writing systems.

Glyph The rendered and therefore visual representation of one or more Unicode characters that can be identified by its code point(s). A glyph is rendered in a specific font in a specific context. No matter how different it looks to the user, for Unicode all different representations of one code point are exactly the same. Sometimes one character is represented as two glyphs.

Unicode code points are often organized in blocks. A block can, for example, contain all letters of the Latin script. Those blocks are helpful although not always consistent. The IPA is represented in a basic block but many IPA symbols are actually found in other blocks. Confusion often arises from the fact, that one human-perceived letter/character/grapheme is sometimes represented as more than one code point.

Grapheme clusters A grapheme cluster is one visual letter that is represented as more than one code point. This is the case for diacritic marks. Note that sometimes, these can be precomposed and the combination of those two or more characters is assigned a new number. These clusters can be problematic if in a specific context, the graphemes should not be clustered but read separately. Unicode has ways to solve this but it is still important to be aware of it.

Additional complexity is added through the possibility of Unicode to create Unicode locales. These allow users to specify language- or writing-system-specific cases. An additional challange is that of picking the right font. Our standard font format can only contain about half of all the Unicode code point. It is therefore simply not possible to display the entire set of Unicode characters with one font. Many problems encountered with displaying writing systems are somehow connected to

the font rather than Unicode itself [Moran and Cysouw, 2018]. Moran and Cysouw [2018] list a few more 'pitfalls' that one might encounter when dealing with Unicode.

For the present thesis, this topic is relevant for two reasons:

1. The IPA contains many special characters and many diacritics.
2. The language data is available in many different scripts.

It is crucial that all data files, be it phonetic or 'normal' scripts, are formatted and read correctly.

3.6 Representing phonemes

In order to input phonemes to any kind of mathematics-based model, they need to be represented numerically. Instead of representing the phoneme directly it is possible to represent one phoneme as a feature vector. This idea is not new as there is the idea that we can represent words as a vector of numbers which we then input to a model. Each number in that vector is what is called a *feature*. **?** note that a problem of such featurization is that the words are no longer comparable across different models as the words are possibly featurized in different ways. This is why the present a tool called *WordKit* which is a Python library that allows to featurize words in a standardized way.

There are models that make use of such features vectors. Tan example of feature data can be found here.

3.7 Low-resource setting

Apart from a few well-studied examples, for most languages there is only little available data. It is therefore highly interesting and important to find solutions of how to deal with lack of data. Hammond [2021] submitted a system to the 2021 SIGMORPHON edition focusing on data augmentation methods. The primary goal of their approach was to test how successful a minimalist data augmentation model would be, knowing it would most probably not outperform any of the other models. They identified two approaches that might improve low-resource models. The first one is to use as much as possible of the development set for training. The second to train all languages together differentiating the languages only by a tag added to the word representations. The model they used was purposefully a very simple model

that does not use a lot of resources. They used a seq2seq neural net with a LSTM decoder and encoder. Both LSTMs have two levels.

3.8 Random background

I put things here that I had to look up working on this thesis but that might no end up in the final thesis, it is basically my notebook. I might put in into the glossary if it makes sense... Monte Carlo simulation: a simulation that evolves randomly. We can, for example, estimate pi with a Monte Carlo simulation. The most basic intuition is that I can estimate something from random samples. The important thing is that the selection of the samples must be random and cannot be biased. A second factor that influences the reliability of the results is that the sample size must be large enough. According to the law of large numbers this is a common rule when estimating numbers. The Monte Carlo simulation can be used in situations where it is not possible to explore all possible combinations that are needed to produce a certain outcome (e.g. measuring the hight of all people living on this planet to obtain the average). In such cases we can pick a large enough sample randomly.

source

Student's t-test: This is a statistical test that tells us something about the significance of the difference between one result compared to another. If I have two averages over two related (paired) or unrelated (unpaired) groups, the t-test tells me if the change from one average to the other is statistically significant. If not, the change can occur just as well by chance.

There are systems that can produce speech directly from orthography and question the necessity of phonetic transcriptions. When only little data is available, the training data might not be enough to train a orthography-to-phoneme system, making phonetic transcriptions necessary. Another reason for creating phonetic transcriptions is that it usage is not limited to speech applications [Mortensen et al., 2018]. They might also be used to compare languages on speech basis. In order to do that, there needs to be a lot of knowledge about how language works. Comparing languages and studying their similarities and differences is part of a well-established branch of traditional linguistics called comparative linguistics. The analysis of large amounts of text in any language is commonly referred to as corpus linguistics. Corpus linguistics allows for both qualitative and quantitative analysis of text. Although text can refer to written or spoken language, most corpora contain written text [McEnery and Hardie, 2011]. Multilingual corpora can be used to compare

languages. If all of these different approaches are combined, we end up by what we could call comparative corpus phonetics.

Add quick intro into corpus linguistics, quantitative analysis, this is essentially what is done with the corpus. [McEnery and Hardie, 2011]

Introduction to comparative linguistics at some place. [Hock and Joseph, 2019]

4 Data Collection

The first important part of this thesis is concerned with data collection. Although phonetics is an important sub-area in linguistics, phonetic transcriptions are hard to find. If there are any transcriptions available, there are various hindrances that prevent it from being used as is. The following chapter outlines the different data types which are available and the different strategies that are used to convert the data into one well-formatted corpus. Apart from hindrances concerning sources and format, there are issues concerning the data itself. There are generally many more different pronunciations of a word than there are spellings. It is thus important to specify clearly what dialect or pronunciation convention a phonetic transcription follows.

4.1 Transcription Conventions

Another problem that needs be dealt with are different transcription conventions. There are different phonetic languages and within those there are different levels of transcription details. The most common are listed below.

IPA The IPA has one of the most common phonetic transcription conventions used in linguistics.

DISC The DISC convention is different from most of the others as it assigns exactly one ASCII code to each phone. The alphabet covers only Dutch, English and German phone inventories [R H. Baayen and Gulikers, 2021]. It is therefore very impractical for a multilingual corpus. However, it is still in use and can be found in some papers (e.g. Rao et al. [2015]).

In order to guarantee comparability, some transcriptions need to be translated into other transcription conventions.

Apart from different character sets there are different levels of detail. Not all transcriptions represent the phonetics in equal detail. Generally, there is the distinction of broad and narrow transcription. These two go back to the linguistic distinction

of phone and phoneme. Broad refers to a phonemic description. Following the linguistic definition in chapter 4, this means that the transcription does not transcribe speaker specific pronunciations or dialectal variations. This kind of transcription is therefore less complex and usually easier to create and understand. Narrow transcriptions are phonetic. They present every speaker individual or dialectal sounds as exactly as possible. Although the spoken text in narrow and broad transcription sounds only minimally different, the two texts can diverge greatly. It is important to treat broad and narrow transcriptions as two different kinds of transcriptions.

(4.1) pɪˈkʰ

(4.2) pɪˈkʰ

Example 4.1 is a narrow (phonetic) transcription of the beginning of the Mapudungun version of the short story *The North Wind and the Sun*. The same text is transcribed broadly (phonemic) in example 4.2. As becomes clear in this example, the narrow transcriptions is much longer as it contains more different characters. The problem, with especially the narrow transcriptions, is that the transcriber still needs to define what narrow means in a specific case. This becomes tricky when given a task to automatically transcribe text, the training data might employ one definition of narrow, while there are texts in the test set that might follow another definition. However, in practice data is very rare, so in the end you would probably just use any data you can get.

4.2 Transcription Sources & Formats

Phonetic transcriptions of various languages are available from different sources in different formats. In order to use those, they have to be converted into simple text format in appropriate encoding that can easily be read and processed by a machine. The following subsections list the different data types and how they are used.

4.2.1 Full Text

For the task at hand, phonetic transcriptions in the form of fully transcribed texts would be ideal. As became clear, it is hardly possible to find those. There is plenty of material describing how different languages can be transcribed but those rarely contain fully transcribed text. If they do, it is mostly limited to one or a few sentences. The JIPA continuously published different phonetic transcriptions of a

short story called “The North Wind and the Sun”. A collection of those is available in a handbook of the JIPA which is only available as a pdf scan of the original book [Press, 2010]. While OCR is technically possible it turns out to be very difficult for IPA characters. The tools that exist do sometimes include IPA character recognition like the ABBYY FineReader which can be acquired for a fee. The CL institute at the UZH owns a version of the ABBYY tool but this version does not include the IPA module although ABBYY generally supports IPA character recognition. This ABBYY version was run on a JIPA pdf containing said phonetic transcriptions but the result could not be used. Mostly diacritics and special phonetic symbols were not correctly transcribed. There are also open source tools. One of which is called tesseract. tesseract does not include the IPA alphabet. It is possible to train the model to include the IPA alphabet but this would need appropriate training data. Add quote

Some transcriptions have been published in separate issues as part of a collection of articles called ”Illustrations of the IPA”. While some of them are available in plain text format most of them are only available as pdfs or even images in text books. It is of course possible to manually type-write those which is what I did. More on how this is best done is explained in chapter 5 on experiments. Table 2 shows the languages for which the short story is available and which are also in the corpus.

Iso639-3	Type	Variation	Language
arn	broad and narrow		Mapudungun
cmn		Pekinese	Mandarin Chinese
deu	broad and narrow	North German	German
ell			Modern Greek
eng	broad and narrow		English
eus	broad and narrow	Goizueta	Basque
hau	narrow		Hausa
heb			Modern Hebrew
hin	narrow		Hindi
ind			Indonesian
kat	broad and narrow		Georgian
kor			Korean
mya			Burmese
pes			Western Farsi
spa	broad and narrow	Castilian	Spanish
tha			Thai
tur	broad	Istanbul	Turkish

Table 2: The table shows a list of all the short stories “The North Wind and the Sun” that are available as phonetic text and whose languages are in the corpus.

Additionally, some texts include short descriptions where certain pronunciations rules are explained which are not included in the transcriptions (especially stress).

4.2.2 Pronunciation Dictionaries

Another data type that is found quite often are lists of words' pronunciation. Those are sometimes referred to as pronunciation dictionaries. However, these often mean that there are words mapped to an audio representation which is not what is meant in this present case. Pronunciation dictionary in this present case refers to the mapping of an orthographic word to its pronunciation using phonetic symbols. Although such lists are very handy, especially as they can easily be used to train a transcription model, transcriptions of individual words and of entire texts are not exactly the same. There are two major problems:

- Pronunciation depends on the context of the word in question. Word forms are ambiguous and sometimes their pronunciation differs given on their specific context. add example
- Phonetic boundaries are not always equivalent with word boundaries. Spoken language sometimes merges certain words which leads to one phonetic unit. There are phonetic symbols to represent such merging which often happens in, for example, French.

WikiPron

There exist databases of pronunciation dictionaries. Many of those do not release the mining software used to extend the database with more languages [Lee et al., 2020]. A very recent project that publishes pronunciation lists is WikiPron. The WikiPron project [Lee et al., 2020] is an open-source Python mining tool to retrieve pronunciation data from Wiktionary. Their database contains 1.7 million word/pronunciation pairs in 165 languages. Both, the database and the tool, are freely available online. Apart from the mining tool and the database, WikiPron can be used for grapheme-to-phoneme modelling. More on this subject will be discussed in chapter **??**. In both G2P shared tasks organized by SIGMORPHON (see **??**, data provided by WikiPron was used. For the 2021 task, WikiPron was improved and additional scripts were added based on feedback and findings in the 2020 task. One major improvement was concerned with languages written in different scripts. WikiPron supports now the detection of different scripts and languages can be sorted according to those scripts.

5 Experiments

This chapter presents the experiments and practical explorations that I conducted for this thesis. The previous chapters listed the different steps and problems that arise when trying to create and analyse a phonetic corpus.

5.1 Typewriting pdf phonetic transcriptions

In order to make use of as much data as possible, I used a software to manually transcribe the pdf scans. The software allows to make use of neural Handwritten Text Recognition (HTR) models. There exists no pre-trained IPA model but I trained my own while transcribing the documents. On the website they mention that ideally training needs 5,000 - 10,000 words already transcribed. Although my available data is not nearly enough to train a reliable model, it was a great help to transcribe. As the scans where not handwritten text, the model still reached a surprisingly good quality. For the Hebrew transcription, the model reached a WER of 34.52% and a CER of 6.11%. The two main mistakes were made for two characters that were not even in the training data. The quality of the scans differed quite a lot which had an influence on the performance of the model as well. After transcribing another document I trained the model again and transcribed the remaining documents. The transcriptions got continuously better such that in the end for the last documents I did not take me nearly as much time as in the beginning. Most of the errors resulted from characters that had not been in the previously described documents. I did not run a closer analysis so this is only my intuition.

Transkribus allows to use public models and share their own. Technically, I can share my model as well. It needs to be clarified whether it is actually possible as there is not a lot of training data involved and the models performance differs.

potentially add short table for WER and CER values for a few languages

5.2 Pronunciation dictionary coverage

In order to get an understanding of how many words are covered in the word lists, I created a script to calculate the coverage, the WER and the CER. I replaced the words in the texts with the words in the word list and compared it to the reference transcription. While dealing with the full texts and the word lists, I noticed several things that are important when dealing with those texts.

- The pronunciation dictionaries sometimes included duplicates with different pronunciations. This is not surprising but still it needs to be handled well. A solution is to simply delete duplicate words. A close examination also showed that sometimes, the duplicate pronunciations are wrong. As it is the case with the English word "would". add this example indented
- For some full texts it is not clear whether their transcription is narrow or broad. On the other hand, sometimes there is no broad or narrow word list available for a specific language but only one of those. In order to find out how similar broad and narrow texts and word lists are, the calculations were run for every possible combination of each language. For languages that had both types of text and both types of word lists, the calculations were run four times.
- The IPA allows to transcribe intonation segments. In German, those correspond mostly to punctuation marks like end of sentence symbols or commas. But this must not be true for every case. It needs to be decided if those should be kept or potentially deleted.
- In order to do this very simple experiment, it is necessary to tokenize the texts. This works well for languages using the Latin script. For languages like Chinese or Korean this is more difficult to accomplish. However, this issue needs to be tackled to create G2P models anyway. I will therefore not explore this issue here.
- WikiPron filtered some datasets for the SIGMORPHON shared task. verify that. whenever available I used the filtered version.

The results from this experiment are summarized in table 3. Generally it is good to see, that most texts are at least partially covered by the pronunciation dictionary. A closer examination of the results shows a few language specific issues that might be relevant in further experiments.

Chinese Although the Chinese coverage is rather high, the WER is very bad. This is due to the fact, that in the lists the tones are represented differently than in the reference text. It needs to be analysed if one of these formats can be converted into another format. check the different tone transcriptions

Hebrew The only language that is not covered at all is Hebrew. A closer examination showed that the words in the text have many diacritics, while the words in the list do not have many diacritics. Additionally, the list is very short.

Whenever there were four experiments per language, the combination of the broad reference text written with the broad list had the best WER and CER or in the case of English the best WER and a slightly worse CER. However, this finding needs to be analysed with caution as the narrow word lists contain always less words than the broad ones. Interestingly enough, sometimes it does not matter if the text written with the broad word list is compared to the narrow transcription or the broad. In fact, for English, the text written with the broad list compared to the narrow reference shows a better CER. This suggests that the differences in broad and narrow transcriptions are not great. For languages where the type of the transcriptions was unclear but two lists for available, the broad list produced the better results if there was any difference.

5.3 Automatic G2P

As there is only very little data available as full texts, we decided to use the short stories as test set for the experiments with G2P models. Those texts are all manually created and are specifically created by linguists for the purpose of studying phonetics of many languages.

5.3.1 Training settings

There are different settings which I will used to train the models.

Setting 1: Baseline Small Test all languages with the CMU model where we have data available. We want to have a baseline for all languages. All models are trained separately. Broad and narrow transcriptions are treated as separate languages and thus trained separately as well. The same is true for dialects if there is any information available. American and British English are trained in different

Iso 639-3	Coverage	WER	CER	Type ref	Type list	Num words list
cmn	87.5	2.3	0.84		broad	133 686
deu	75.0	0.77	0.52	broad	broad	34 145
deu	22.22	0.98	0.82	narrow	narrow	10 984
deu	75.0	0.85	0.52	narrow	broad	34 145
deu	22.22	0.98	0.83	broad	narrow	10 984
ell	6.14	1.0	0.9		narrow	408
ell	22.81	0.94	0.85		broad	10 547
eng	92.04	0.92	0.38	broad	broad	57 230
eng	7.08	1.01	0.83	narrow	narrow	1 633
eng	7.08	1.0	0.83	broad	narrow	1 633
eng	92.04	0.92	0.36	narrow	broad	57 230
eus	5.75	0.97	0.85	broad	broad	1 742
eus	0.0	1.0	0.92	narrow	narrow	186
eus	0.0	1.0	0.89	broad	narrow	186
eus	5.75	0.98	0.89	narrow	broad	1 742
heb	0.0	1.28	0.92		broad	1 439
heb	0.0	1.28	0.92		narrow	146
ind	22.22	0.97	0.8		broad	1 555
ind	1.85	1.0	0.91		narrow	2 637
kat	43.66	0.86	0.66	broad	broad	15 123
mya	7.14	0.98	0.93	broad	broad	4 631
spa	64.95	0.51	0.41	broad	broad	60 677
spa	35.05	0.99	0.69	narrow	narrow	52 190
spa	35.05	1.0	0.62	broad	narrow	52 190
spa	64.95	0.84	0.56	narrow	broad	60 677
tha	20.0	1.0	0.98		broad	15 050
tur	18.46	1.0	0.91		broad	1 789
tur	6.15	1.0	0.96		narrow	1 812
hin	31.78	1.03	0.69		narrow	9 563
hin	57.36	0.93	0.52		broad	10 812
kor	18.64	1.0	0.96		narrow	14 141
pes	50.0	1.09	0.75		broad	6 128
pes	18.0	1.1	0.91		narrow	1 922

Table 3: The table shows the coverage, WER and CER when the pronunciation dictionaries are used to write “The North Wind and the Sun”.

models as well. I used the WikiPron pronunciation dictionaries to train the model. While some of the data has been down-sampled in the shared task, I used all that was available. Whenever a filtered version was available I used that one. The model is trained with the least amount of effort. Default settings are used and no hyperparameters are changed. The model is trained for the minimum number of

steps which is 10,000 in this case.

Setting 2: Baseline Large This setting is similar to setting 1 except that the model is trained as long as possible for each language. All models have been trained for 200,000 steps and the default settings.

5.3.2 Preprocessing

Before any of the data can be used, it needs to be preprocessed. As I am going to use the pronunciation dictionaries which consist of single words per line with their pronunciation, the full texts need to be prepared like that as well. I did the following steps to convert the full short story texts into pronunciation dictionaries:

- Conversion of || to ‖. Some transcriptions include the double vertical line to mark a major intonation groups in the text. In some transcriptions this is a written as two single vertical lines. Those were replaced by the former to be consistent.
- Removal of ties bars in the phonetic transcriptions. Tie bars are not adding any valuable information. get quote for this!
- Removal of suprasegmentals, except long and half long mark, and the extra short mark.

5.3.3 g2p-seq2seq CMUSphinx

I tested a git repo where they make an already pretrained G2P model available. Setting it up was not very easy as there were issues with the tensorflow version and some other dependencies. Also, they do have a pre-trained model, but this uses a completely different transcription convention than IPA. So we cannot use this model. But to test the model and have a baseline, we trained in on the data we have, to see how it performs. First, I trained it on those languages where I have results from the SIGMORPHON 2021 challenge. The results are compared in table 5.

5.4 Things to include and sort out later

Data Stats In order to get a feeling of the data and what it covers, I collected phoneme and grapheme profiles of the data and compared it to the Phoible dataset.

ISO396-3	BS WER	BS WER short stories	SIG21 WER	Notes
eng (us)	54.40	87.60	37.43	broad
fra	7.20	47.20	5.11	broad
ell	9.80	83.20	18.67	broad
kat	0.30	65.20	0.00	broad
hin	5.60	87.10	5.11	broad
jpn	6.60	-	4.89	narrow
kor	28.70	100.00	16.20	narrow
vie	7.50	100.00	0.89	narrow

Table 4: Baseline CMUSphinx results compared with SIGMORPHON 2021 results. The table shows the results for setting 1. CMUSphinx provides a WER implementation which has been used to evaluate the models.

ISO396-3	BS WER	BS WER short stories	SIG21 WER	Notes
eng (us)	50.70		37.43	broad
fra	5.30		5.11	broad
ell	7.10		18.67	broad
kat	0.00		0.00	broad
hin	4.40		5.11	broad
jpn	6.50		4.89	narrow
kor	23.40		16.20	narrow
vie	7.10		0.89	narrow

Table 5: Baseline CMUSphinx results compared with SIGMORPHON 2021 results. The table shows the results for setting 2. CMUSphinx provides a WER implementation which has been used to evaluate the models.

For each language that I am working with, I have two different types of data: first the short story and second the WikiPron G2P dictionary. For each language and each data type I got three lists:

- Grapheme list: contains all graphemes in that language. Characters that need a base character like diacritics are shown together with their base character.
- Phoneme list: contains all phonemes in that language. Again, diacritics and similar characters are shown with their base characters.
- Phoneme cluster list: Phonemes can be clustered into bigger sound groups. How to do this, is an ongoing discussion, but I used the segments library to

get the clusters (compare Moran and Cysouw [2018])

Having those overview for the characters for each language allowed me to compare the character vocabulary to the characters or character clusters available in the Phoible dataset. This comparison showed that quite a few characters are missing that are included in the WikiPron data and the short stories. Some problems and potential strategies to solve it:

- No real IPA: There are sometimes characters included that are no part of the IPA. There might be a reason why the authors of the transcriptions decided to use this special character to denote a particular sound, but this is not always known. A possibility is to try and map it to a character that is available in Phoible and that represents a similar sound (or even the same sound actually).
- Tie bars: The creators of Phoible decided to exclude tie bars because they add no real value to the transcriptions. add source for this?
- Stress marks: Stress marks are not represented in Phoible as they do not represent a sound.
- Tones: Even within the IPA there exist different conventions of how to represent tones. Some are better suited for different languages. Apart from different ways of representing tones, it is not always sensible to have tones represented. Mostly, tones are not written as speakers of that language know how to pronounce the tones. So, the questions is whether it is necessary to include the tones at all. When looking at the written representation it does not matter what the tones are as the basic phonemes do not change. This is of course different when the phonetic representation is mapped to a spoken representation. add more on that, maybe in background chap

References

100-language-sample. WALS Online - Languages. In M. S. Dryer and M. Haspelmath, editors, *The World Atlas of Language Structures Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig, 2013. URL `https://wals.info/languoid/samples/100`.

R. Aharoni and Y. Goldberg. Morphological Inflection Generation with Hard Monotonic Attention, 2016. URL `https://arxiv.org/pdf/1611.01487`.

L. F. Ashby, T. M. Bartley, S. Clematide, L. Del Signore, C. Gibson, K. Gorman, Y. Lee-Sikka, P. Makarov, A. Malanoski, S. Miller, O. Ortiz, R. Raff, A. Sengupta, B. Seo, Y. Spektor, and W. Yan. Results of the Second SIGMORPHON Shared Task on Multilingual Grapheme-to-Phoneme Conversion. In *Proceedings of the 18th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, Stroudsburg, PA, USA, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.sigmorphon-1.13.

L. Baird, N. Evans, and S. J. Greenhill. Blowing in the wind: Using 'north wind and the sun' texts to sample phoneme inventories. *Journal of the International Phonetic Association*, page 1–42, 2021. doi: 10.1017/S002510032000033X.

M. Bisani and H. Ney. Joint-sequence models for grapheme-to-phoneme conversion. *Speech Communication*, 50(5):434–451, 2008. ISSN 0167-6393. doi: https://doi.org/10.1016/j.specom.2008.01.002. URL `https://www.sciencedirect.com/science/article/pii/S0167639308000046`.

E. Chodroff. Corpus Phonetics Tutorial, 2019. URL `https://eleanorchodroff.com/tutorial/index.html#`.

S. Clematide and P. Makarov. CLUZH at SIGMORPHON 2021 Shared Task on Multilingual Grapheme-to-Phoneme Conversion: Variations on a Baseline. Association for Computational Linguistics, 2021. doi: 10.18653/v1/2021.sigmorphon-1.17.

B. Comrie, M. S. Dryer, D. Gil, and M. Haspelmath. Introduction. In M. S. Dryer and M. Haspelmath, editors, *The World Atlas of Language Structures Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig, 2013. URL https://wals.info/chapter/s1.

CrashCourse, 2021a. URL https://www.youtube.com/watch?v=vyea8Ph9BOM.

CrashCourse, 2021b. URL https://www.youtube.com/watch?v=-sUUWyo4RZQ&list=PL8dPuuaLjXtP5mp25nStsuDzk2blncJDW&index=18.

V. Gautam, W. Li, Z. Mahmood, F. Mailhot, S. Nadig, R. Wang, and N. Zhang. Avengers, ensemble! benefits of ensembling in grapheme-to-phoneme prediction. In *Proceedings of the 18th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 141–147, 01 2021. doi: 10.18653/v1/2021.sigmorphon-1.16.

K. Gorman, L. F. Ashby, A. Goyzueta, A. McCarthy, S. Wu, and D. You. The SIGMORPHON 2020 shared task on multilingual grapheme-to-phoneme conversion. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 40–50, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.sigmorphon-1.2. URL https://aclanthology.org/2020.sigmorphon-1.2.

M. Hammond. Data augmentation for low-resource grapheme-to-phoneme mapping. In *Proceedings of the 18th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 126–130, Online, Aug. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.sigmorphon-1.14. URL https://aclanthology.org/2021.sigmorphon-1.14.

H. H. Hock and B. D. Joseph. *Language History, Language Change, and Language Relationship*. De Gruyter, 2019. ISBN 9783110613285. doi: 10.1515/9783110613285.

N. Jaitly, D. Sussillo, Q. V. Le, O. Vinyals, I. Sutskever, and S. Bengio. A neural transducer, 2016.

M. Kracht. *Introduction to linguistics*. Los Angeles, 2007. URL https://linguistics.ucla.edu/people/kracht/courses/ling20-fall07/ling-intro.pdf.

J. L. Lee, L. F. Ashby, M. E. Garza, Y. Lee-Sikka, S. Miller, A. Wong, A. D. McCarthy, and K. Gorman. Massively Multilingual Pronunciation Modeling with WikiPron. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4223–4228, Marseille, France, 2020. European Language Resources Association. ISBN 979-10-95546-34-4. URL https://aclanthology.org/2020.lrec-1.521.

M. Y. Liberman. Corpus Phonetics. *Annual Review of Linguistics*, 5(1):91–107, 2019. ISSN 2333-9683. doi: 10.1146/annurev-linguistics-011516-033830.

R. Y.-H. Lo and G. Nicolai. Linguistic knowledge in multilingual grapheme-to-phoneme conversion. In *Proceedings of the 18th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 131–140, Online, Aug. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.sigmorphon-1.15. URL https://aclanthology.org/2021.sigmorphon-1.15.

P. Makarov and S. Clematide. CLUZH at SIGMORPHON 2020 shared task on multilingual grapheme-to-phoneme conversion. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 171–176, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.sigmorphon-1.19. URL https://aclanthology.org/2020.sigmorphon-1.19.

T. McEnery and A. Hardie. *Corpus Linguistics: Method, theory and practice*. Cambridge textbooks in linguistics. Cambridge University Press, Cambridge, 2011. ISBN 9780511981395. doi: 10.1017/CBO9780511981395.

S. Moran and M. Cysouw. *The Unicode Cookbook for Linguists: Managing writing systems using orthography profiles*. 06 2018. ISBN 978-3-96110-090-3. doi: 10.5281/zenodo.1296780.

D. R. Mortensen, S. Dalmia, and P. Littell. Epitran: Precision G2P for many languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May 2018. European Language Resources Association (ELRA). URL https://aclanthology.org/L18-1429.

B. Peters and A. F. T. Martins. One-size-fits-all multilingual models. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 63–69, Online, July 2020. Association for Computational Linguistics. doi:

10.18653/v1/2020.sigmorphon-1.4. URL https://aclanthology.org/2020.sigmorphon-1.4.

C. U. Press. The principles of the international phonetic association (1949). *Journal of the International Phonetic Association*, 40(3):299–358, 2010. doi: 10.1017/S0025100311000089.

R. P. R H. Baayen and L. Gulikers, 2021. URL https://catalog.ldc.upenn.edu/docs/LDC96L14/.

K. Rao, F. Peng, H. Sak, and F. Beaufays. Grapheme-to-phoneme conversion using long short-term memory recurrent neural networks. *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4225–4229, 2015.

Sigmorphon. SIGMORPHON - Special Interest Group on Computational Morphology and Phonology, 2021. URL https://sigmorphon.github.io/.

SPUR project. Non-randomness in Morphological Diversity, 2021. URL https://www.spur.uzh.ch/en/departments/research/textgroup/MorphDiv.html.

S.-A. Toma and D. Munteanu. Rule-based automatic phonetic transcription for the romanian language. *Future Computing, Service Computation, Cognitive, Adaptive, Content, Patterns, Computation World*, 0:682–686, 11 2009. doi: 10.1109/ComputationWorld.2009.59.

X. Yu, N. T. Vu, and J. Kuhn. Ensemble self-training for low-resource languages: Grapheme-to-phoneme conversion and morphological inflection. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 70–78, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.sigmorphon-1.5. URL https://aclanthology.org/2020.sigmorphon-1.5.

A Tables

Table 6: The table shows a list of the 100 languages in the corpu
mation on the language families.

Iso639-3	Name WALS	
39-3abkWALSAbkhazWALSNorthwest Caucasian6639-3	–WALS	
6639-3	–WALS	–V
6639-3	–WALS	
6639-3	–WALS	–
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	–WAL
6639-3	–WALS	
6639-3	–WALS	–
6639-3	–WALS	–WALS39-3
6639-3	–WALS	
6639-3	–WALS	–WALS
6639-3	–WALS	–WALS
6639-3	–WALS	
6639-3	–WALS	–WA
6639-3	–WALS	–WALS
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	–WALS
6639-3	–WALS	
6639-3	–WALS	

Iso639-3	Name WALS	
6639-3	–WALS	–W
6639-3	–WALS	–WALS
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	–WALS39
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	–
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	–WALS39-
6639-3	–WALS	–W
6639-3	–WALS	–WALS39-3dniWALSD
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	–WALS
6639-3	–WALS	
6639-3	–WALS	–
6639-3	–WALS	
6639-3	–WALS	–WAL
6639-3	–WALS	
6639-3	–WALS	–WALS39
6639-3	–WALS	–WALS
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	

Iso639-3	Name WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	–WALS39-3m
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	–V
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	–V
6639-3	–WALS	–V
6639-3	–WALS	–WA
6639-3	–WALS	
6639-3	–WALS	–W
6639-3	–WALS	
6639-3	–WALS	–
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	
6639-3	–WALS	

Iso639-3	**Name WALS**
6639-3	–WALS

Iso639-3	Name WALS	Family WALS	
39-3abkWALSAbkhazWALSNorthwest Caucasian6639-3		–WALS	–WALS39-3amp
6639-3		–WALS	–WALS39-3aeyWALSAm
6639-3		–WALS	–WALS39-3apuW
6639-3		–WALS	–WALS39-3bmiWALSB
6639-3		–WALS	–WALS39-3bsnWA
6639-3		–WALS	–WALS39-3gryWA
6639-3		–WALS	–WALS39-3eus
6639-3		–WALS	–WALS39-3apeWALSArape
6639-3		–WALS	–WALS39-3bskWALSI
6639-3		–WALS	–WALS39-3ramWALSC
6639-3		–WALS	–WALS39-3tzmWALSBerber (Mi
6639-3		–WALS	–WALS39-3chaWALSC
6639-3		–WALS	–WALS39-3cktWALSChukchiV
6639-3		–WALS	–WALS39-3zocWALSZoque (C
6639-3		–WALS	–WALS39-3
6639-3		–WALS	–WALS39-3haeWALSOron
6639-3		–WALS	–WALS39-3arzWALSArabic
6639-3		–WALS	–WALS39-3fijWA
6639-3		–WALS	–WALS39-3
6639-3		–WALS	–WALS39-3gniWAL
6639-3		–WALS	–WALS39-3k
6639-3		–WALS	–WALS39-3hauW
6639-3		–WALS	–WALS39-3hebWALSHebrew
6639-3		–WALS	–WALS39-3hinWAL
6639-3		–WALS	–WALS39-3hixWA
6639-3		–WALS	–WALS39-3hnjWALSHm
6639-3		–WALS	–WALS39-3qviWALSQuechua
6639-3		–WALS	–WALS39-3im
6639-3		–WALS	–WALS39-3indWALSInd
6639-3		–WALS	–WALS39-3kalWALSGreenland
6639-3		–WALS	–WALS39-3
6639-3		–WALS	–WALS39-3gydW
6639-3		–WALS	–WALS39-3kioWAL
6639-3		–WALS	–WALS39-3ckuWA
6639-3		–WALS	–WALS39-3ko
6639-3		–WALS	–WALS39-3sesWALSKoy
6639-3		–WALS	–WALS39-3k
6639-3		–WALS	–WALS39-3ku
6639-3		–WALS	–WALS39-3lk
6639-3		–WALS	–WALS39-3lajWALSL
6639-3		–WALS	–WALS39-3lvkWALSLavukaleveW
6639-3		–WALS	–WALS39-3lezWALSLezgia
6639-3		–WALS	–WALS39-3dniWALSDani (Lower Grand Val