



Article

A Survey of Grapheme-to-Phoneme Conversion Methods

Shiyang Cheng ¹, Pengcheng Zhu ², Jueting Liu ^{2,*} and Zehua Wang ³

¹ School of Environment and Spatial Informatics, China University of Mining and Technology, No.1 Daxue Road, Xuzhou 221000, China; csy1402@cumt.edu.cn

² School of Computer Science and Technology, China University of Mining and Technology, No.1 Daxue Road, Xuzhou 221000, China; ts22170035a31@cumt.edu.cn

³ Department of Electrical and Computer Engineering, University of British Columbia, 2332 Main Mall, Vancouver, BC V6T 1Z4, Canada; zwang@ece.ubc.ca

* Correspondence: 6476@cumt.edu.cn

Abstract: Grapheme-to-phoneme conversion (G2P) is the task of converting letters (grapheme sequences) into their pronunciations (phoneme sequences). It plays a crucial role in natural language processing, text-to-speech synthesis, and automatic speech recognition systems. This paper provides a systematical overview of the G2P conversion from different perspectives. The conversion methods are first presented in the paper; detailed discussions are conducted on methods based on deep learning technology. For each method, the key ideas, advantages, disadvantages, and representative models are summarized. This paper then mentioned the learning strategies and multilingual G2P conversions. Finally, this paper summarized the commonly used monolingual and multilingual datasets, including Mandarin, Japanese, Arabic, etc. Two tables illustrated the performance of various methods with relative datasets. After making a general overall of G2P conversion, this paper concluded with the current issues and the future directions of deep learning-based G2P conversion.

Keywords: grapheme-to-phoneme conversion; speech synthesis; machine learning; deep learning



Citation: Cheng, S.; Zhu, P.; Liu, J.; Wang, Z. A Survey of Grapheme-to-Phoneme Conversion Methods. *Appl. Sci.* **2024**, *14*, 11790. <https://doi.org/10.3390/app142411790>

Academic Editor: Douglas O'Shaughnessy

Received: 10 November 2024

Revised: 7 December 2024

Accepted: 12 December 2024

Published: 17 December 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Natural Language Processing (NLP) is a crucial branch of artificial intelligence aimed at enabling computers to understand, generate, and process human language. Text-to-speech (TTS) technology is a significant application of NLP and is widely used in voice assistants, accessibility communication, and other fields. In TTS systems, the grapheme-to-phoneme (G2P) model converts text into phoneme sequences, directly impacting the quality of speech synthesis and determining whether TTS systems can achieve state-of-the-art performance levels. G2P is a language processing technology that maps graphemes (such as letters or letter combinations) in a writing system to their corresponding phonemes, the smallest units of speech. This technology converts letter strings like *APPLE* into phoneme strings like *[AEPAHL]*. Formally, let Σ_G be the grapheme alphabet and Σ_P be the phoneme alphabet. For a word w in a language, grapheme-to-phoneme refers to the mapping:

$$g(w) \rightarrow p(w), \quad (1)$$

where $g(w) \in \Sigma_G^*$ and $p(w) \in \Sigma_P^*$ are the grapheme and phoneme representations of w , respectively.

Grapheme-to-phoneme conversion has a wide range of applications and plays a crucial role in text-to-speech and automatic speech recognition systems. The state-of-the-art performance of these systems depends heavily on the accuracy of the grapheme-to-phoneme conversion. In text-to-speech systems, a high-quality G2P model is an essential component that significantly impacts the system's quality. Inaccurate grapheme-to-phoneme conversion can lead to unnatural pronunciations and unintelligible speech. Given the inherent complexity of languages, the G2P task presents numerous challenges. Initially, the

many-to-one and one-to-many mapping relationships between graphemes and phonemes, coupled with the contextual dependency of pronunciation, significantly augment the task's complexity. Subsequently, the pronounced disparities in spelling and pronunciation rules across various languages and dialects, which are exacerbated by the introduction of loanwords, further complicate the G2P conversion process. In the context of multilingual and low-resource language conversion tasks, G2P conversion within a multilingual framework necessitates the accommodation of the divergent spelling and phoneme systems of different languages, while low-resource languages grapple with issues of data scarcity.

Numerous researchers have proposed many solutions for the grapheme-to-phoneme conversion task. From a developmental perspective, grapheme-to-phoneme conversion algorithms can be categorized into traditional rule-based and dictionary-based methods, statistical machine learning methods, and deep learning-based conversion methods.

Very early grapheme-to-phoneme conversion systems were specifically designed for English, utilizing language-specific characteristics or domain knowledge to formulate rules aimed at addressing the variations within the English language. These systems required custom-tailored rules for individual languages, and their performance was largely confined to the language itself. Dictionary-based methods also had many limitations; even frequent updates to the dictionary could not guarantee coverage of all words in a given language.

The combination of dictionaries and rules offered relatively better performance, whether they primarily used a dictionary with post-processing rules or primarily used rules for languages with regular pronunciations while only using dictionary lookups for exceptional cases. This traditional approach could accurately convert input text into corresponding phonemes. Overall, these methods achieved good recognition results in specific corpora, but they were costly to develop and could not be easily ported to other languages. Even in highly regular languages, there are special cases, especially for loanwords from foreign languages, where rule-based methods could lead to incorrect conversions.

Statistical machine learning methods surpassed rule-based methods, alleviating some of the burden of technical expertise and improving model performance, but they required training and tuning to achieve optimal performance.

Neural network models eliminated the dependency on explicit alignment and captured finer and more effective features, and it could generalize well to predict the correct pronunciations of unseen words, thus achieving high performance. The introduction of attention mechanisms addressed the issue of neural networks not being able to compute in parallel. Even with long data sequences, attention mechanisms could capture focal information without losing important details, reducing model parameters, speeding up computation, and improving model performance. Due to the similarities in vocabulary, spelling, writing systems, and phoneme inventories between low-resource and high-resource languages, multilingual MT techniques have been used to train large-scale multilingual grapheme-to-phoneme conversion systems.

Besides discussion about the various deep-learning models, this paper also represents four learning strategies for G2P conversion. A pre-trained model is a kind of deep learning model that has been trained; the pre-trained model can be fine-tuned to adapt to other specific tasks. Transfer learning is the idea of applying features learned from one problem to a new, similar problem. Multi-task learning is a form of collaborative learning where multiple tasks are learned in parallel, and the results influence each other. Multimodal learning aims to integrate and process different types of data during the learning process.

Although the majority of G2P converters are monolingual, multilingual G2P conversion is also a popular research direction since the grapheme-to-phoneme conversion is a subfield of linguistics. After talking about the learning strategies, this paper discussed recent approaches for multilingual G2P conversion.

The above Figure 1 illustrates the overall structure of this paper. This paper will mainly discuss the various methods of G2P conversion and focus on deep learning-based G2P; then, the learning strategies and multilingual G2P will be covered; finally, this paper will talk about the dataset and the evaluation metrics.

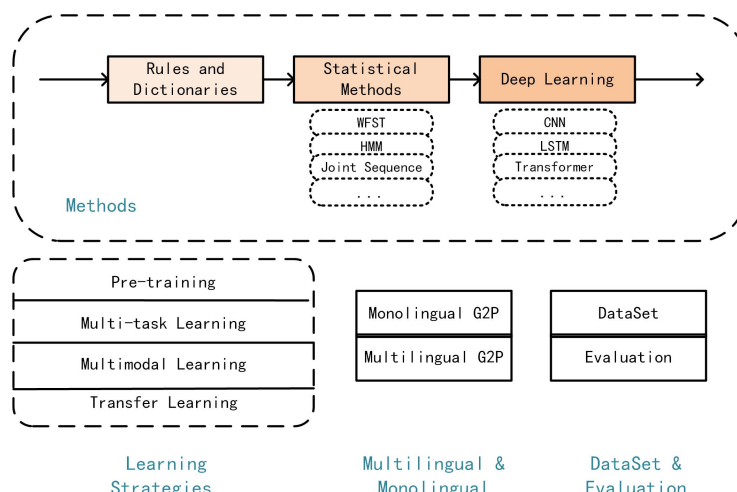


Figure 1. Overall structure of this survey.

2. Grapheme-to-Phoneme Methods

Figure 2 illustrates commonly used grapheme-to-phoneme conversion methods. This section will list G2P methods sequentially from the traditional rule and dictionaries to the most recent deep learning-based conversion. This paper will focus on deep-learning methods.

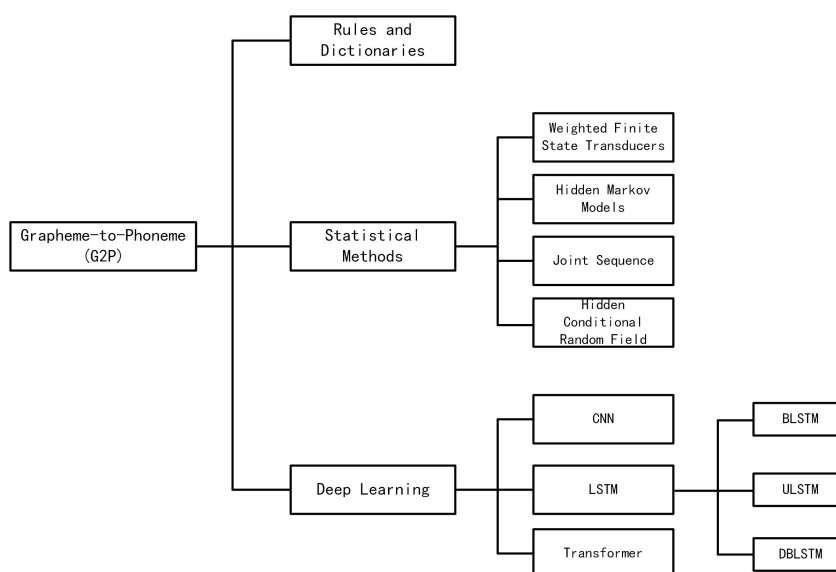


Figure 2. Commonly employed grapheme-to-phoneme conversion methods.

2.1. Traditional Methods

2.1.1. Rule-Based and Dictionary-Based Methods

The simplest method for grapheme-to-phoneme conversion is dictionary lookup, but constructing a large dictionary is both expensive and time-consuming, and there will always be words outside the vocabulary. Therefore, rule-based methods were proposed to address the issues with dictionary lookups. Rule-based grapheme-to-phoneme conversion is a traditional language processing technique that requires mapping the graphemes (shapes and arrangements of letters) to their corresponding phonemes (the smallest units of speech) through predefined sets of rules. This method typically relies on the in-depth understanding of language structure and pronunciation rules by linguists or domain experts. By summarizing the spelling and pronunciation rules of a language, phonetic conversion rules are manually formed to predict the pronunciation of words. Developing such a grapheme-to-phoneme system requires the use of numerous grapheme-to-phoneme rules

and expert knowledge. For certain languages (such as Chinese and Japanese) with complex writing systems, constructing rules requires a significant amount of human effort and is costly, and it is difficult to cover most possible scenarios. The literature [1] proposed a rule-based G2P system, formulating a set of 329 grapheme-to-sound rules to convert English text into the International Phonetic Alphabet (IPA). The literature [2] studied mathematical and computational tools for reasoning about regular languages and regular relations, proposing several rule-based methods. However, some languages (such as Chinese and Japanese) have very complex writing systems, requiring a significant amount of human effort to construct grapheme-to-phoneme rules, and there is no guarantee that the dictionary will contain all words in the text.

The literature [3] used a hand-seeded algorithm to achieve optimal alignment, then employed decision tree technology to implement a general framework for building letter-to-sound (LTS) rules from a list of words in a language, successfully constructing LTS models for four different languages. This method can remove regular examples from the word list, allowing even smaller dictionaries to cover all words. The model's performance in handling unknown words was evaluated in detail through the conversion results of unknown words in news texts. The model can generate reasonable pronunciations for unseen words, with errors mainly concentrated on proper nouns, especially those from foreign languages.

The literature [4] proposed a method to represent the complexity of languages to guide developers in addressing the question of "how to use words to learn rules", designing a word selection strategy that outperforms random selection. Moreover, the introduction of knowledge about word alignment and rule order errors can further improve this strategy. However, this method is not suitable for languages with unclear connections between graphemes and phonemes, as the number of grapheme-to-phoneme conversion rules in these languages almost grows linearly with the size of the dictionary.

2.1.2. Statistical Machine Learning-Based Methods

Statistical machine learning-based methods utilize probabilistic statistics and machine learning algorithms to build a speech model with sufficient training data and employ decoding algorithms to label each word.

This type of task can be summarized as follows: given a sequence of letters L , find the phoneme sequence P^* that maximizes $\Pr(P|L)$,

$$P^* = \arg \max_P \Pr(P | L) = \arg \max_P \Pr(L, P) \quad (2)$$

Selecting either the conditional distribution $\Pr(P|L)$ or the joint distribution $\Pr(L, P)$ can accomplish this task; both frameworks have seen extensive work on grapheme-to-phoneme conversion. N-gram models have been successfully used in speech recognition and other data sequence statistical modeling applications, offering flexibility, compactness, and fast decoding capabilities. Joint models do not require pre-aligned data for training and are a natural way to generate the alignment needed for conditional models. Additionally, joint models are typically symmetric, allowing them to be used directly for both grapheme-to-phoneme and phoneme-to-grapheme conversion. In the literature [5], the Expectation Maximization (EM) Algorithm is utilized to infer the minimal correspondence between the graphemes and phonemes; the G2P converter is constructed by a joint N-gram model. Experimental results also show that adding accent information and word frequency information to the training data can improve the model's accuracy. Formally, within the context of a joint sequence model, consider a word w with its spelling represented as $\gamma(w) = l_1 l_2 \dots l_n$, where l_i denotes a letter, and its pronunciation represented as $\pi(w) = p_1 p_2 \dots p_m$, where p_i denotes a phoneme. The joint probability $P(Y(w), \pi(w))$

is pertinent in this scenario. The task can be articulated as identifying the most probable transcription π^* for the given input γ :

$$\pi^* = \arg \max P(\gamma, \pi) \quad (3)$$

Here, $P(\gamma, \pi)$ signifies the joint probability of the spelling γ and the pronunciation π . Ref. [6] combined the joint n -gram model with a decision tree model to measure the impact of improved G2P models on LVCSR performance in challenging ASR tasks, proposing the use of N best pronunciation variants instead of a single best pronunciation.

Ref. [7] first applied Weighted Finite State Transducers (WFST) to the G2P conversion task, using EM for alignment. This effectively reduced storage space and accelerated decoding speed, significantly shortening training time compared to [8] without degrading performance. Ref. [9] introduced a new open-source WFST-driven G2P conversion toolkit that supports a range of features, including EM-driven many-to-many alignment algorithms and three novel decoding techniques. The training process involves first converting aligned sequence pairs into aligned joint label pairs, $(g1 : p1, g2 : p2, \dots, gn : pn)$, training a joint N -gram model, and then using the mitlm tool to convert the N -gram model into a WFST. Ref. [10] improved joint sequence modeling by enhancing the alignment algorithm and using an RNN LM alongside the standard n -gram LM for N -best rescoring.

Ref. [11] investigated two related issues in the WFST-based G2P conversion domain. The first was the impact of the method used to convert target words into equivalent finite-state machines on downstream decoding efficiency. The second issue was the impact of the method used to represent the joint n -gram model within the WFST framework on system speed and accuracy.

Ref. [12] applied Hidden Markov Models (HMMs) to G2P transformation, and [13] modeled context at HMMs' observations with discrete observations and discrete probability distributions. Ref. [14] proposed a new many-to-many alignment training technique. Bigram prediction of letter chunks automatically managed digraphs and diphones, combining HMM methods with local classification models to predict the global phoneme sequence given a word, achieving good results.

Ref. [15] proposed two methods for integrating alignment into CRF training processes. The first method uses a two-step approach with maximum approximation, iteratively improving the initial alignment. The second method considers all possible alignments given certain constraints.

Ref. [16] proposed Hidden Conditional Random Field (HCRF) models, which use hidden variables to model the alignment of grapheme and phoneme sequences. Ref. [17] compared three methods for defining hidden conditional random fields that can express a wide range of mappings between grapheme and phoneme sequences. At the cost of performance degradation, they introduced a model inspired by the phrase-based machine translation framework, which significantly reduced computational costs by using efficient methods to prune the search space. Ref. [18] combined the joint n -gram model with the CRF model, enhancing the generalization capability of the model.

2.2. Methods Based on Deep Learning

With the development of artificial intelligence, more researchers are using deep learning models to tackle the G2P conversion problem. Their goal is to outperform earlier approaches in terms of accuracy and make it simple to include in speech recognition and end-to-end text-to-speech systems. Compared with the traditional statistical machine learning approaches, neural network-based G2P conversion models are usually more robust with better performance. However, they have to face common problems in deep learning such as overfitting, vanishing and exploding gradients, and high computational resource requirements.

2.2.1. LSTM

Long Short-Term Memory Networks (LSTM) have been successfully applied to tasks such as speech labels for acoustic frames, handwriting recognition, and language modeling. In acoustic frame labeling tasks, LSTM outperforms standard Recurrent Neural Networks (RNN). The grapheme-to-phoneme conversion task, which involves transcribing sequences from words to pronunciations, requires temporal context and is well-suited for LSTM network structures.

Ref. [19] proposed a general end-to-end sequence learning method that utilizes Long Short-Term Memory networks (LSTM) to map input sequences to vectors of a specific dimension, and then uses another deep LSTM to decode the target sequence from this vector, eliminating the need for separate alignment between grapheme and phoneme sequences.

Inspired by the model in [19], ref. [20] first applied LSTM to G2P conversion, investigating grapheme-to-phoneme models based on Long Short-Term Memory networks. They proposed three models: ULSTM, DBLSTM, and a combined model.

The ULSTM model featured 1024 memory cells in the output layer, with a softmax activation function and cross-entropy loss function, initialized with random weights. Since ULSTM only utilizes left/past context, the authors introduced the concept of output delay and experimented with Zero-delay, Fixed-delay, and Full-delay scenarios. When using fixed delay, increasing the delay size slightly improved model performance. The Full-delay model outperformed any fixed delay model, achieving a 9.1% PER, demonstrating that larger context information can enhance model performance.

The DBLSTM model is essentially the same as the unidirectional model but includes a “backward” LSTM layer, allowing it to see the entire input before producing the output phonemes. The structure is illustrated in the Figure 3. A single input layer is fully connected to two parallel layers, each with 512 units; one is unidirectional, and the other is bidirectional. The first hidden layer is fully connected to a unidirectional layer containing 128 units, which is then connected to the final output layer. The model is initialized with random weights and trained with a learning rate of 0.01. Connectionist Temporal Classification (CTC) is used as the output layer. The CTC output layer has a softmax output layer with 41 units, each corresponding to 40 output phoneme labels, plus an additional blank unit, where the probability of the blank unit indicates no label observed at a given time step.

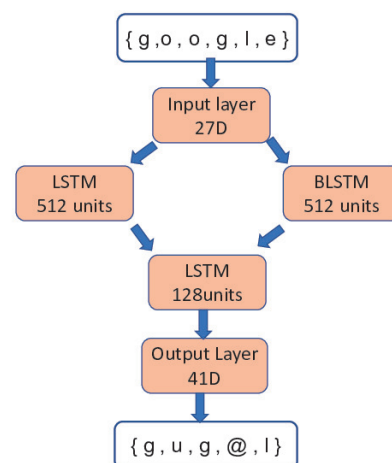


Figure 3. DBLSTM-CTC Model.

The performance of the BLSTM architecture outperforms that of the unidirectional LSTM and is roughly equivalent in accuracy to the joint n-gram model but with fewer parameters and faster speed. LSTM models are built at the full sequence (word) level, while joint n-gram models are built at the grapheme level. The authors also explored the performance of combining LSTM-based modeling methods with joint n-gram modeling

methods. The output of the LSTM model is represented as a Finite State Transducer (FST), which is then intersected with the FST output from the n-gram model, selecting a single best path in the final generated FST. This combined model achieved very good results on the CMU dataset, with a word error rate of 21.3%, demonstrating that combining the two modeling methods can produce a better overall model.

The end-to-end sequence learning method proposed by [19] shows excellent performance. Due to the use of alignment, the method proposed by [8] achieved excellent conversion results. Inspired by this, ref. [21] employed alignment-based models in their research on G2P. The model structure is illustrated in Figure 4.

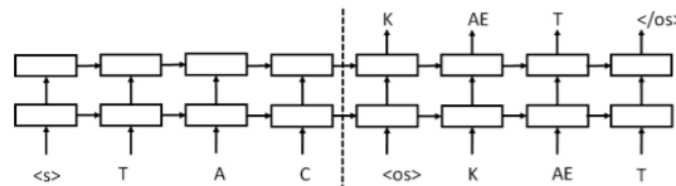


Figure 4. Encoder–decoder LSTM.

The LSTM network structure is straightforward and can effectively capture the intricate relationships between letter sequences and phoneme sequences. One of its advantages is that it does not require explicit alignment information. The LSTM is trained using Backpropagation Through Time (BPTT), and the beam search solver selects the hypothesis sequence with the highest posterior probability as the decoding result, generating the phoneme sequence during the decoding phase. Although the encode-decode model does not require alignment, using alignment information to construct the model yields better results. The model structure is illustrated in Figure 5.

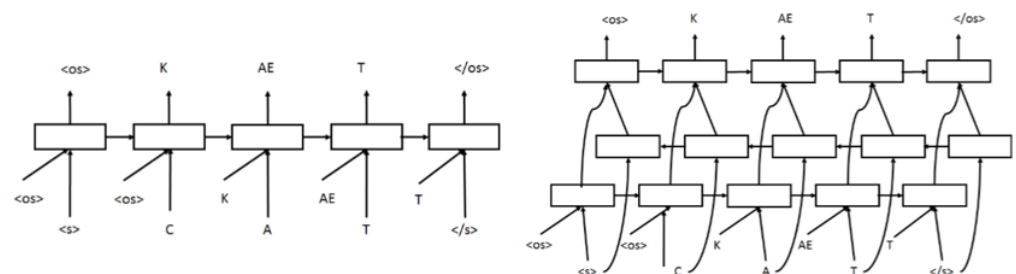


Figure 5. Alignment-based model.

The authors investigated alignment-based models, finding that unidirectional LSTM, unable to observe the entire input sequence, has a higher Word Error Rate (WER) with a default window size. The bidirectional LSTM (Bi-LSTM) model demonstrated strong performance, achieving a Phoneme Error Rate (PER) of 5.45% and a Word Error Rate (WER) of 23.55%. The model’s prediction performance improved when it could observe the entire input sequence.

In recent G2P models, alignment issues have not received much attention. Ref. [20] completely avoided alignment, while [21] simplified alignment to 1-to-1 or 1-to-2 cases. This did not maximize the performance of LSTM-based G2P models. Ref. [22] proposed an improved method for G2P models based on deep bidirectional Long Short-Term Memory (BLSTM) RNNs, utilizing complex many-to-many G2P alignments to enhance G2P model performance. They applied deep bidirectional LSTM with hyperparameter optimization, investigating models with varying numbers of hidden layers, linear projection layers, input concatenation windows, and different alignment constraints. The model was enhanced with

grapheme-level LMs based on unidirectional LSTM, backoff models, and MAXENT models. Further improvements were achieved by decoding BLSTM outputs using a beam search mechanism with a phoneme-level LM. The best model, optimized with hyperparameters, achieved a Phoneme Error Rate (PER) of 5.37% and a Word Error Rate (WER) of 23.23%.

Building on the encoder–decoder architecture developed by [21], ref. [23] used a multi-layer bidirectional encoder with Gated Recurrent Unit (GRU) nonlinearities and a similarly deep unidirectional GRU decoder, where each decoder layer’s initial state was initialized to the corresponding encoder forward layer’s final hidden state. The model was trained with teacher forcing and decoded using beam search, achieving 5.8% PER and 28.7% WER on CMUDict.

Most G2P research uses pronunciation dictionaries as input, and training models to predict phonemes word by word, making it difficult to handle cross-word assimilation effects. Ref. [24] constructed a G2P converter based on deep neural networks, using short word sequences for training, which could easily handle (in a language-independent manner) inter-word and cross-word effects in phoneme sequences. The model was validated on three different languages (English, Russian, and Czech) and could easily be adapted with a set of transcription rules for irregular and regular pronunciation languages, competing with traditional dictionary and rule-based baseline methods in all tested languages. The model structure is illustrated in Figure 6.

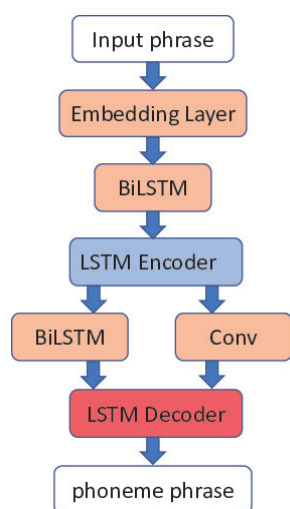


Figure 6. The structure of DNN-based converter for G2P.

The model architecture is illustrated in the figure below: This model uses phrases as inputs to the network, converting all words W_i in the phrase into symbol sequences of equal length by using padding symbols. The first layer of the model is the embedding layer, which converts each symbol/grapheme into a vector representation; and then embed it into a bidirectional LSTM layer to learn the phonetic relationships within words; the next LSTM encoder layer creates word embeddings to generate fixed-length representations for each word, with the sequence dimension changing from characters to words, and then passes these word embeddings to the next layer—biLSTM layer or convolutional layer; these layers learn the cross-word-boundary relationship; finally, an LSTM decoder is utilized to decode the fixed-length intermediate representation into phonemes. The decoder generates a posterior probability sequence p_i of phonemes from word representations. The BiLSTMDNN model consists of two neural sequences, one for the positive time direction and the other for the negative time direction. This model has the ability to learn both past and future states simultaneously, introducing convolutional layers between the encoder and decoder. The model focuses on the direct neighboring context of specific words.

Many methods cannot achieve both speed and performance, ref. [20] proposed the CTC-based model requires output length to implement a delay strategy, and the large

model parameters limit the application scenarios. Thus, ref. [25] integrated the advantages of expert knowledge and the CTC algorithm and proposed a fast, lightweight, and theoretically parallel new model LITEG2P, which can be applied simultaneously on both cloud and device, with accuracy comparable to models based on Transformers, but with a speed increase of over 30 times.

2.2.2. CNN

Convolutional neural network (CNN), due to its strong ability to extract internal representations between objects that have spatial relationships, is widely utilized in various fields. Besides LSTM models, research made efforts to develop G2P conversion models with CNN. Ref. [26] proposed a non-sequential greedy decoding (NSGD) model and combined fully convolutional encoder–decoder model with NSGD, compared with the recurrent neural networks, CNN can outperform in non-sequential tasks.

LSTM sequentially reads inputs and outputs depending on the previous input, LSTM cannot parallelize these networks while applying CNN can reduce computational complexity by using a large receiving domain. Ref. [27] first applied CNN in G2P conversion and proposed a Seq2Seq G2P model based on BiLSTM and CNN. The CNN layer of this method uses pixels as inputs for convolution operations, introduces residual connections to solve the problem of gradient vanishing, increases the number of network layers, and improves model performance.

The G2P conversion of convolutional neural networks based on residual connections (mid part in Figure 7) has high PER (5.84%) and WER (29.74%), but the model has fewer parameters. The encoder convolutional neural network and decoder Bi LSTM (right part in Figure 7) model with residual connections achieved the best PER (4.81%) and WER (25.13%), with PER better than previous solutions on both CMU and NetTalk datasets, and WER very close to the results obtained in [28].

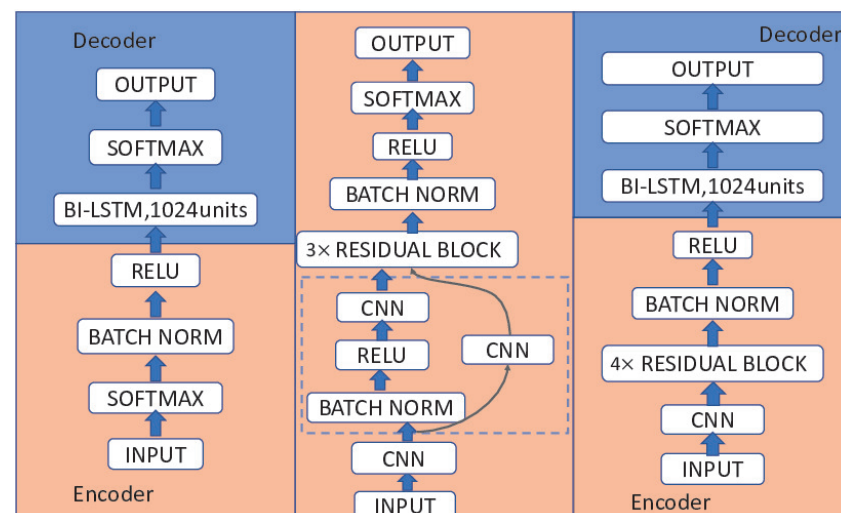


Figure 7. The structure of CNN and BiLSTM G2P converters.

2.2.3. Transformer

The Transformer model consists of an encoder and a decoder, both of which are composed of multiple stacked self-attention layers and fully connected layers. The attention mechanism is an attention mechanism that associates different positions of a single sequence to calculate its internal representation. Add positional encoding to input and output embeddings without using any loop layers. Location information provides the order of input and output sequences for Transformer networks. Ref. [29] utilized the attention mechanism in English-German translation that achieved a state-of-the-art result. The Transformer network structure is illustrated in Figure 8.

Ref. [30] first introduces the Transformer to G2P conversion to obtain better performance. Ref. [31] proposed a transformer structure for grapheme-to-phoneme conversion, which improves the performance of the model by increasing the number of encoder-decoder layers. This work studied 3×3 (3-layer encoder and 3-layer decoder), 4×4 , and 5×5 models based on Transformer and evaluated them. The 4×4 model achieved the best performance, with a PER of 5.23% and a WER of 22.1%, which illustrates appropriately increasing the number of encoder and decoder layers can improve model performance, but the model will have relatively more training parameters; a large number of parameters will increase the training time of the model and may also reduce its performance. The model structure is illustrated in Figure 8.

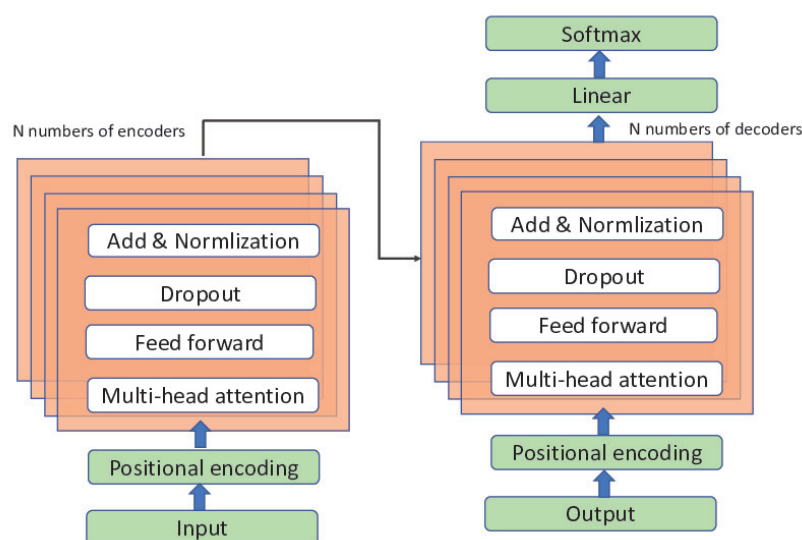


Figure 8. Transformer structure.

Compared with other deep learning G2P models, the model with the best performance in Transformer has competitiveness, small model parameters, short training time, fewer training resources required, and higher accuracy. Ref. [32] proposed A novel Transformer G2P model architecture (NN-KoG2P) for the Korean language, which reflects the characteristics of Korean pronunciation and has fast inference speed. It can generate pronunciation sequences in real-time services.

3. Strategies of G2P Conversion

Besides deep learning-based methods for achieving grapheme-to-phoneme conversion, this paper also lists a number of strategies for deep learning-based G2P, including pre-training, multi-task learning, multimodal learning, and transfer learning.

3.1. Pre-Training for G2P Conversion

Pre-training is a process in which a model is initially trained on a large or general dataset. Then, the model for a specific task can be fine-tuned by the corresponding dataset. Ref. [33] fine-tuned a G2P conversion model from a pre-trained Text-to-Text Transfer Transformer (T5) model named T5G2P. This model carried out English and Czech tasks which is able to convert an input text sentence into a phoneme sequence. The subsequent works include [34,35] that the researchers expanded their pre-trained G2P model to multiple languages. However, most of the pretraining models rely on large-scale datasets, in G2P field, these corpora may not apply to some languages. Inspired by the BERT language model, ref. [36] proposed a pre-trained grapheme model named "grapheme BERT" (GBERT), which is trained in a self-supervised manner on large word lists of specific languages. GBERT aims to capture the contextual relationships among graphemes within words. Experimental

results demonstrate that GBERT-based G2P models perform exceptionally well under both medium-resource and low-resource data conditions. The model structure is illustrated in Figure 9.

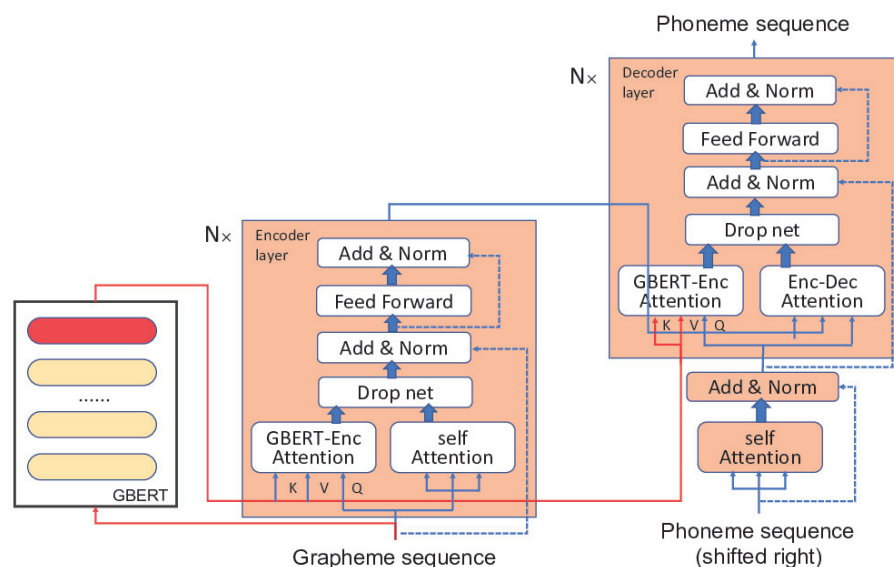


Figure 9. GBERT pre-trained G2P.

3.2. Multi-Task Learning for G2P Conversion

MTL (Multi-Task Learning) is a method of modeling human inductive transfer concepts as machine learning models. It enhances the generalization capability and learning efficiency of individual tasks by simultaneously learning multiple related tasks. By leveraging the interdependencies among these tasks, MTL shares underlying features, which more effectively utilizes data and reduces overfitting compared to single-task learning. This approach is particularly strong in areas like natural language processing, where text contains various prompts that facilitate the completion of multiple tasks simultaneously. MTL helps focus the model's attention on truly important features, making it highly suitable for NLP applications.

Ref. [37] trained a speech-language model in a multitasking training environment. The perplexity of the joint model in all languages is lower than that of individual speech-language models. Inspired by this work, ref. [38] studied using MTL to improve the performance of the Seq2Seq G2P converter and trained a single Seq2Seq model on a polyphonic dictionary dataset containing multiple languages and pinyin letters. However, this method has not shown improved error rates in multilingual learning. Ref. [39] presented a multi-task sequence to sequence training to enhance the English pronunciation generation of the German G2P model. Improve English pronunciation recognition results by generating English pronunciations and adding them to the pronunciation dictionary of the ASR model. The improvement of the English word test set indicates that this method has the potential to solve the challenge of English words in German ASR.

3.3. Multimodal Learning for G2P Conversion

The majority of existing G2P systems are trained separately on spelling and pronunciation data, ignoring audio modalities. IPA character sequences used to represent pronunciation do not capture all of their subtle differences. Ref. [40] raised a multimodal model with an attention mechanism. This model can learn robust joint representations of grapheme and speech data in multiple languages and then predict phoneme sequences given only grapheme. Speech data are only used for training, and the model only utilizes grapheme input during inference. The model structure is illustrated in Figure 10.

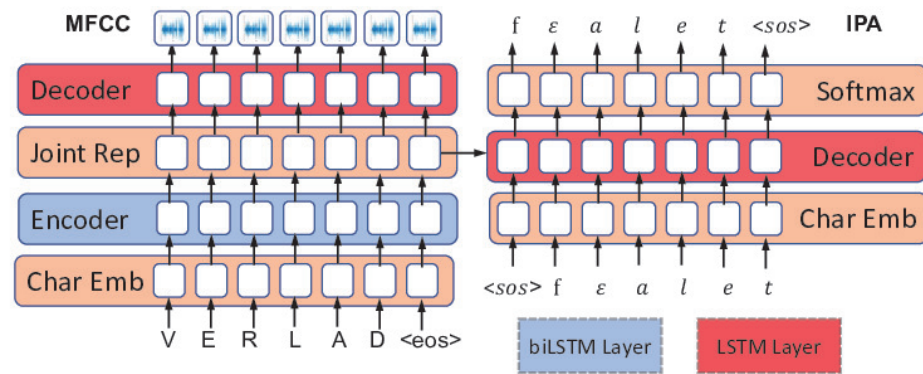


Figure 10. Multimodal G2P.

The model is a Seq2Seq model with a single encoder and two decoders; one is the MFCC decoder composed of an LSTM, which predicts MFCC coefficients from grapheme (sub-task), and the other decoder task is to predict IPA character sequences (main task). This multimodal model performs well in both accuracy and efficiency, reducing the phoneme error rate from 8.21% to 6.39% compared with the single modal. However, this study still relies on spelling and pronunciation pairs, which means that this method may not be applicable to languages without spelling and pronunciation pairs.

3.4. Transfer Learning for G2P Conversion

For low-resource languages, a single language dataset cannot meet the requirements of the Transformer model. The G2P system supports multiple languages and can be trained on several languages datasets to improve data availability in low-resource languages [41].

Ref. [42] utilized transfer learning to construct an end-to-end multilingual grapheme-to-phoneme model to improve the conversion performance of foreign words in different languages. The author combines the model with corresponding input parameters, embeds language distribution (Bidage ID) and language labels (System ID) (such as <en US>, <en UK>, <de DE>, <fr FR>, etc.) as input embeddings, connects them with attention context vectors, and feeds them forward to the decoder’s fully connected layer before softmax. The language distribution vector represents the degree to which a specific word belongs to each language, and the calculation formula is as follows:

$$p(id | l) = \frac{C(w)}{\log |N|}, \quad \sum_l p(id | l) = 1 \quad (4)$$

where N refers to the size of the dictionary, $C(w)$ refers to the count of occurrences of words found in a given language dictionary. The language ID vector helps the model distinguish words with multiple pronunciations in different languages and model their phoneme distribution accordingly. The performance of this work shows that for low-resource languages, the average phoneme error rate decreased by 7.2%.

Transformer-based models require a large amount of training data and are not suitable for dialects with limited data. Ref. [43] adopts a transfer learning-based approach to cross-learn English dialects and establishes a G2P model based on Transformer, achieving high accuracy in dialect conversion with small available data. The model structure is illustrated in Figure 11.

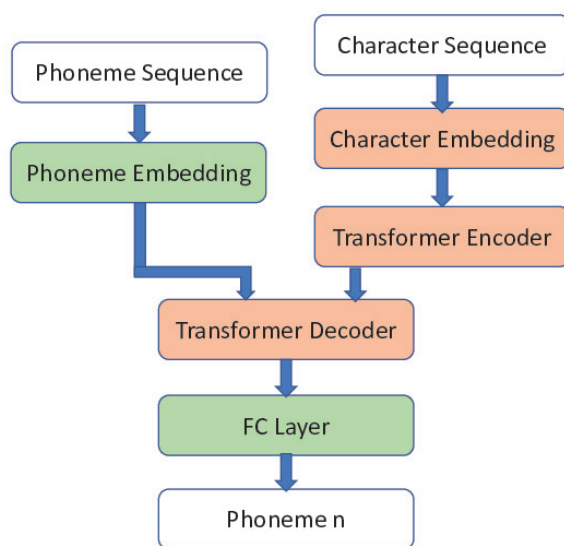


Figure 11. Transfer learning for G2P system.

Ref. [44] combined the global character vector GCV and bidirectional recurrent neural network BRNN to establish a high-performance multilingual G2P model, which enables the model to process multiple languages simultaneously. And further explored multilingual G2P and monolingual G2P in multiple languages.

These studies suggest that transfer learning is an effective approach to building powerful G2P models for dialects with limited data. For low-resource languages, using pre-trained models for transfer learning can significantly improve PER and quality, while increasing dictionary size can improve model performance. Using dictionaries and generated phonemes to fine-tune the model can also improve its accuracy. The drawback is that pre-trained models require a large amount of training data for transfer learning between different dialects or languages.

4. Multilingual G2P Conversion

A monolingual G2P converter means that it is necessary to train a model for each language. On the other hand, the multilingual approach proposes using a single model to handle two or more languages.

Ref. [45] utilized the multilingual speech dictionary of Wikipedia and the rule table of Wikipedia to establish a high-resource G2P model and proved that adding G2P rules as training data can improve the performance of the converter. The author trained weighted finite state transition (WFST) models on various high-resource languages and then transferred these models to low-resource languages. The language distance metric was used to select which high-resource models to use, and the phoneme distance metric was used to map the phonemes of high-resource languages to the phoneme library of low-resource languages. The G2P model for high speech resources was improved, and 229 G2P models for low-resource languages were established. To overcome the scarcity of data in low-resource languages. Ref. [46] proposed a Seq2Seq neural network model that uses spelling pronunciation training from hundreds of languages to train the G2P model for multilingual conversion. The model includes a global attention architecture and a 2-layer Bilstm-based encoder and decoder, utilizing the dataset collected from [45].

The establishment and maintenance costs of pronunciation dictionaries are high, and free high-quality dictionaries are only suitable for a few languages, which limits the development of G2P for some languages. Ref. [47] developed an open-source tool WikiPron, the authors extracted pronunciation data from Wiktionary and created a pronunciation database for 165 languages, removing stress and syllable boundary markers. They used pronunciation strings encoded with the International Phonetic Alphabet and segmented

them through the segments library. Ref. [48] presented a multilingual grapheme-to-phoneme conversion quest with a dataset, evaluation metrics, and strong baselines. Nine participating groups submitted 23 G2P systems, compared with the Seq2Seq baseline; this reduces the word error rate by up to 18%.

Ref. [49] proposed a rule-based lightweight and accurate G2P system Epiran, which maps orthographic data to the speech space through consistent grapheme and phoneme mapping. It supports 61 languages and can be easily extended to new languages. Compared to dictionaries, Epiran has no vocabulary limit and can recognize words outside the dictionary used to train the model. It can generate consistent and accurate pronunciations for all words encountered by the system. In low-resource languages, the relationship between orthography and pronunciation may not be clear, while Epiran can provide accurate G2P conversion methods for some low-resource languages. However, evaluating a multilingual model like Epiran is difficult as each supported language must be effectively evaluated individually.

Ref. [50] raised a multilingual G2P model based on attention mechanism Transformer architecture, using byte-level input representation to adapt to different grapheme systems. Byte representation has a smaller vocabulary compared to traditional character representation, reducing lexical sparsity and thus improving the accuracy of the model. In the development of byte-level models, ref. [51] conducted further work and proposed a NART-CRF structure for fast bilingual G2P real-time processing.

Due to the lack of multilingual pronunciation dictionaries to train multilingual models, encoding a large number of correct labels in the model remains challenging. Ref. [52] presented a G2P model based on BYT5. The author implemented a G2P model based on BYT5 (a pre-trained model) to solve large-scale multilingual G2P conversion problems. The error rate is lower than that of monolingual models. This work demonstrates that the byte-level model BYT5 can outperform most monolingual G2P conversion models. Ref. [53] proposed a multilingual phoneme recognizer, Allophant, which can be applied to low-resource languages or even zero sample languages. This architecture combines speech embedding with a multi-task speech attribute classifier, improving the ability to convert invisible language. The above research can illustrate that for low-resource languages and zero-shot tasks, the deep learning-based multilingual G2P conversion will have better performance compared with the monolingual G2P converters.

5. Dataset and Evaluation Metrics

5.1. Dataset

5.1.1. Monolingual Dataset

The most widely used English G2P conversion datasets are CMUDict and NetTalk [54]. The CMUDict is an open-source pronouncing dictionary for North American English developed by Carnegie Mellon University, it contains 134,000 words and their pronunciations that are represented in ARPAbet format. NetTalk contains a list of 20,008 English words, with a phonetic transcription for each word, the NetTalk dataset is created for training a proper converter task. Besides the English G2P dataset, there are also various monolingual datasets for different languages. For example, ref. [55] created a Mandarin dataset that consists of 99,000+ sentences for Chinese polyphone disambiguation; ref. [56] proposed a Kurdish text corpus for the Central Kurdish (Sorani) branch; ref. [57] introduced an end-to-end Persian G2P converter with Persian dataset; ref. [58] provided an Arabic phonetics database that contains 46,000 files; ref. [59] described a 150,000-entry Japanese phonetic database; ref. [60] developed a text corpus for the Malayalam language with 62,000 words.

5.1.2. Multilingual Dataset

Most multilingual G2P conversion research chooses to utilize multiple monolingual datasets as the dataset. The lack of a multilingual dictionary is quite a barrier to multilingual G2P conversion. There are still some multilingual datasets. Ref. [47] introduced wikiPron, an extracted pronunciation data from Wiktionary, which contains 1.7 million pronunciations

from 165 languages. Ref. [61] provided a series of dictionaries consisting of wordlists with accompanying phonemic pronunciation information in the International Phonetic Alphabet (IPA). Ref. [52] applied previous datasets and developed a G2P dataset from various sources that covers 100 languages.

5.2. Evaluation Metrics

To evaluate the performance of a G2P converter, Phoneme Error Rate (*PER*) and Word Error Rate (*WER*) are the two most commonly used metrics. Both of these two metrics are used to measure the similarity between the predicted text and the reference text. *WER* is a word-level indicator which can be defined as the rate of word errors in a predicted text:

$$WER = \frac{Substitutions + Deletions + Insertions}{Number\ of\ Words\ in\ Reference} \quad (5)$$

PER is a phoneme-level indicator that can be calculated as the rate of phoneme errors in a predicted text:

$$PER = \frac{Substitutions + Deletions + Insertions}{Number\ of\ Words\ in\ Reference} \quad (6)$$

where the *Substitutions*, *Deletions*, *Insertions* are edit operations that change one text to another reference text.

PER is more concerned with phoneme-level errors and is suitable for evaluating phoneme conversion systems' performance; *WER* is more concerned with word-level errors and is suitable for evaluating speech recognition systems' performance. In general, lower *WER* and *PER* indicate better conversion performance. *WER* is generally considered the primary metric for grapheme-to-phoneme conversion tasks because even a small conversion error can lead to a noticeable decline in the quality of downstream speech applications.

5.3. Comparison of Different G2P Models

Here are two tables of the performance of G2P models summarized from mentioned research papers. Table 1 compares the results of the various current G2P conversion algorithms on the CMUdict dataset and NetTalk Dataset; all the G2p models in the table are designed as English monolingual conversion. Table 2 illustrates the multilingual G2P conversions on different datasets. Since there is no general multilingual grapheme-to-phoneme database, all the datasets for the multilingual G2P models are collected or designed by the researchers, so the performance may vary with the change of dataset.

Table 1. Comparison of G2P conversion algorithms on the CMUdict dataset and NetTalk dataset.

Method	CMUdict Dataset		NetTalk Dataset	
	<i>PER</i> (%)	<i>WER</i> (%)	<i>PER</i> (%)	<i>WER</i> (%)
Joint sequence model [8]	5.88	24.53	8.26	33.67
Joint maximum entropy (ME) n-gram model [62]	5.9	24.7		
Bi-LSTM + Alignment [21]	5.45	23.55	7.38	30.77
Uni-directional LSTM [21]	8.22	32.64		
Encoder–decoder LSTM (2 layers) [21]	7.63	28.61		
Many-to-many alignments with deep BLSTM RNNs [22]	5.37	23.23		
Failure transitions for joint n-gram models and g2p conversion [11]	8.24	33.55	5.85	24.42

Table 1. *Cont.*

Method	CMUdict Dataset		NetTalk Dataset	
	PER (%)	WER (%)	PER (%)	WER (%)
Joint n-gram model [63]	7.0	28.5		
End-to-end CNN (with res. connections) (model4) [27]	5.84	29.74		
Encoder CNN, decoder Bi-LSTM (model5) [27]	4.81	25.13	5.69	30.1
Encoder–decoder LSTM with attention layer (model1) [27]	5.23	28.36		
LSTM with Full-delay [20]	9.1	30.1		
DBLSTM-CTC 512 Units [20]		25.8		
8-gram FST [20]		26.5		
DBLSTM-CTC 512 + 5-gram FST [20]		21.3		
Joint multi-gram + CRF [18]	5.5	23.4		
Combination of sequitur G2P and seq2seq-attention and multitask learning [38]	5.76	24.88		
Encoder–decoder with global attention [28]	5.04	21.69	7.14	29.2
Encoder–Decoder GRU [23]	5.8	28.7		
Transformer 4x4 [31]	5.23	22.1	6.87	29.82
CNN with NSGD [26]	5.58	24.1	6.78	28.45
LiteG2P-medium [25]		24.3		
r-G2P (adv) [64]	5.22	20.14	6.64	28.85
MTL (512×3 , $\lambda = 0.2$) [65]	5.26	22.96		

Table 2. Comparison of G2P conversion algorithms on the multilingual datasets.

Method	Dataset Description	PER (%)	WER (%)
Ensemble model [41]	15 languages from Wikionary	14.83	3.41
Encoder decoder multilingual model [66]	Chinese, Tibetan, English, and Korean, totaling 9620 words.		6.02
Multitask Sequence-to-Sequence Models [38]	German from Phonolex, English from CMUDICT dataset	3.73	17.2
Multimodal and Multilingual model [40]	20 languages from the CMU Wilderness dataset	26	37.87
Multilingual neural G2P model [50]	English, French, Spanish, Japanese, Chinese, total 10 million pairs	4.03	16.14
LangID-All [46]	Test set 507 languages, training set 311 languages	37.85	7.41
DialectalTransformerG2P [43]	Different dialects of English, including American English, Indian English, and British English.	1.457	

Table 2. Cont.

Method	Dataset Description	PER (%)	WER (%)
Unioned model [45]	Wiktionary pronunciation dictionaries for 531 languages	14.70	44.14
NART-CRF based G2P model [51]	Korean and English, including 20,000 sentences for each language	0.43	
ByT5-small [52]	The dataset includes 99 languages, each with over 3000 entries	8.8	25.9
T5G2P [34]	The English dataset contains 128,532 unique sentences, and the Czech dataset contains 442,029 unique sentences.	0.535	0.175

6. Discussion

After reviewing the above grapheme-to-phoneme conversion materials, we concluded the current barriers in the G2P research. First is the uniformity of the language; the performance of the model can be noticeably influenced by the characteristics of the language. For one-to-one correspondence between graphemes and phonemes, such as in Italian, Dutch, and Hindi languages, the G2P models always perform better. On the other hand, in languages with one-to-many and many-to-many mappings between graphemes and phonemes, it is hard to build a G2P converter with high accuracy. Another issue is that for some minority languages, it is quite difficult to collect enough data for training a G2P model. We believe the zero-shot model and transfer learning can be potential solutions to these minority languages.

Furthermore, with the development of the large language model (LLM), There are already relevant studies in using LLM to support G2P conversion [67,68]. We believe the retrieval ability and comprehension ability of the large language models will significantly promote the development of G2P conversion.

Author Contributions: Conceptualization, writing—original, S.C.; writing—original draft preparation, P.Z.; writing—review and editing, J.L.; funding acquisition, writing—review and editing, Z.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported by the Fundamental Research Funds for the Central Universities; the grant number is 2023QN1079.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors on request.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Elovitz, H.; Johnson, R.; McHugh, A.; Shore, J. To-sound rules for automatic translation of English text to phonetics. *IEEE Trans. Acoust. Speech Signal Process.* **1976**, *24*, 446–459. [[CrossRef](#)]
2. Kaplan, R.M.; Kay, M. Regular models of phonological rule systems. *Comput. Linguist.* **1994**, *20*, 331–378.
3. Black, A.W.; Lenzo, K.; Pagel, V. Issues in building general letter to sound rules. In Proceedings of the The Third ESCA/COCOSDA Workshop (ETRW) on Speech Synthesis, Blue Mountains, Australia, 26–29 November 1998.
4. Kominek, J.; Black, A.W. Learning pronunciation dictionaries: Language complexity and word selection strategies. In Proceedings of the Human Language Technology Conference of the NAACL, Main Conference, New York, NY, USA, 4–9 June 2006; pp. 232–239.
5. Galescu, L.; Allen, J.F. Bi-directional conversion between graphemes and phonemes using a joint n-gram model. In Proceedings of the 4th ISCA Tutorial and Research Workshop (ITRW) on Speech Synthesis, Perthshire, Scotland, 29 August–1 September 2001.

6. Hahn, S.; Vozila, P.; Bisani, M. Comparison of grapheme-to-phoneme methods on large pronunciation dictionaries and LVCSR tasks. In Proceedings of the Thirteenth Annual Conference of the International Speech Communication Association, Portland, OR, USA, 9–13 September 2012.
7. Kubo, Yotaro, T.H.; Nakamura, A. A method for structure estimation of weighted finite-state transducers and its application to grapheme-to-phoneme conversion. In Proceedings of the INTERSPEECH, Lyon, France, 25–29 August 2013; pp. 647–651.
8. Bisani, M.; Ney, H. Joint-sequence models for grapheme-to-phoneme conversion. *Speech Commun.* **2008**, *50*, 434–451. [[CrossRef](#)]
9. Novak, J.R.; Minematsu, N.; Hirose, K. WFST-based grapheme-to-phoneme conversion: Open source tools for alignment, model-building and decoding. In Proceedings of the 10th International Workshop on Finite State Methods and Natural Language Processing, Donostia–San Sebastián, Spain, 23–25 July 2012; pp. 45–49.
10. Novak, J.R.; Minematsu, N.; Hirose, K.; Hori, C.; Kashioka, H.; Dixon, P.R. Improving WFST-based G2P Conversion with Alignment Constraints and RNNLM N-best Rescoring. In Proceedings of the Interspeech, Portland, OR, USA, 9–13 September 2012; pp. 2526–2529.
11. Novak, J.R.; Minematsu, N.; Hirose, K. Failure transitions for joint n-gram models and G2p conversion. In Proceedings of the Interspeech, Lyon, France, 25–29 August 2013; pp. 1821–1825.
12. Taylor, P. Hidden Markov models for grapheme to phoneme conversion. In Proceedings of the Ninth European Conference on Speech Communication and Technology, Lisboa, Portugal, 4–8 September 2005; pp. 1973–1976.
13. Ogbureke, K.U.; Cahill, P.; Carson-Berndsen, J. Hidden markov models with context-sensitive observations for grapheme-to-phoneme conversion. In Proceedings of the Eleventh Annual Conference of the International Speech Communication Association, Makuhari, Chiba, Japan, 26–30 September 2010; pp. 1105–1108.
14. Jiampojarn, S.; Kondrak, G.; Sherif, T. Applying many-to-many alignments and hidden markov models to letter-to-phoneme conversion. In Proceedings of the Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference, Rochester, NY, USA, 22–27 April 2007; pp. 372–379.
15. Lehnen, P.; Hahn, S.; Guta, A.; Ney, H. Incorporating alignments into Conditional Random Fields for grapheme to phoneme conversion. In Proceedings of the 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Prague, Czech Republic, 22–27 May 2011; pp. 4916–4919.
16. Lehnen, P.; Hahn, S.; Guta, V.A.; Ney, H. Hidden conditional random fields with m-to-n alignments for grapheme-to-phoneme conversion. In Proceedings of the Thirteenth Annual Conference of the International Speech Communication Association, Portland, OR, USA, 9–13 September 2012; pp. 2554–2557.
17. Lehnen, P.; Allauzen, A.; Lavergne, T.; Yvon, F.; Hahn, S.; Ney, H. Structure learning in hidden conditional random fields for grapheme-to-phoneme conversion. In Proceedings of the Annual Conference of the International Speech Communication Association, Lyon, France, 25–29 August 2013; pp. 2326–2330.
18. Wu, K.; Allauzen, C.; Hall, K.; Riley, M.; Roark, B. Encoding linear models as weighted finite-state transducers. In Proceedings of the INTERSPEECH, Singapore, 14–18 September 2014; pp. 1258–1262.
19. Sutskever, I.; Vinyals, O.; Le, Q.V. Sequence to sequence learning with neural networks. In Proceedings of the 27th International Conference on Neural Information Processing Systems, Cambridge, MA, USA, 8–13 December 2014; pp. 3104–3112.
20. Rao, K.; Peng, F.; Sak, H.; Beaufays, F. Grapheme-to-phoneme conversion using long short-term memory recurrent neural networks. In Proceedings of the 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brisbane, Australia, 19–24 April 2015; pp. 4225–4229.
21. Yao, K.; Zweig, G. Sequence-to-sequence neural net models for grapheme-to-phoneme conversion. *arXiv* **2015**, arXiv:1506.00196.
22. Mousa, A.E.D.; Schuller, B. Deep bidirectional long short-term memory recurrent neural networks for grapheme-to-phoneme conversion utilizing complex many-to-many alignments. In Proceedings of the Interspeech 2016, San Francisco, CA, USA, 8–12 September 2016; pp. 2836–2840.
23. Arık, S.Ö.; Chrzanowski, M.; Coates, A.; Diamos, G.; Gibiansky, A.; Kang, Y.; Li, X.; Miller, J.; Ng, A.; Raiman, J.; et al. Deep voice: Real-time neural text-to-speech. In Proceedings of the International Conference on Machine Learning, Amsterdam, The Netherlands, 7–10 July 2017; pp. 195–204.
24. Juzová, M.; Tihelka, D.; Vít, J. Unified Language-Independent DNN-Based G2P Converter. In Proceedings of the INTERSPEECH, Graz, Austria, 15–19 September 2019; pp. 2085–2089.
25. Wang, C.; Huang, P.; Zou, Y.; Zhang, H.; Liu, S.; Yin, X.; Ma, Z. LiteG2P: A fast, light and high accuracy model for grapheme-to-phoneme conversion. In Proceedings of the ICASSP 2023–2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Rhodes Island, Greece, 4–10 June 2023; pp. 1–5.
26. Chae, M.J.; Park, K.; Bang, J.; Suh, S.; Park, J.; Kim, N.; Park, L. Convolutional sequence to sequence model with non-sequential greedy decoding for grapheme to phoneme conversion. In Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018; pp. 2486–2490.
27. Yolchuyeva, S.; Németh, G.; Gyires-Tóth, B. Grapheme-to-phoneme conversion with convolutional neural networks. *Appl. Sci.* **2019**, *9*, 1143. [[CrossRef](#)]
28. Toshniwal, S.; Livescu, K. Jointly learning to align and convert graphemes to phonemes with neural attention models. In Proceedings of the 2016 IEEE Spoken Language Technology Workshop (SLT), San Diego, CA, USA, 13–16 December 2016; pp. 76–82.

29. Luong, T.; Pham, H.; Manning, C.D. Effective Approaches to Attention-based Neural Machine Translation. *arXiv* **2015**, arXiv:abs/1508.04025.
30. Sun, H.; Tan, X.; Gan, J.W.; Liu, H.; Zhao, S.; Qin, T.; Liu, T.Y. Token-level ensemble distillation for grapheme-to-phoneme conversion. *arXiv* **2019**, arXiv:1904.03446.
31. Yolchuyeva, S.; Németh, G.; Gyires-Tóth, B. Transformer based grapheme-to-phoneme conversion. *arXiv* **2020**, arXiv:2004.06338.
32. Kim, H.Y.; Kim, J.H.; Kim, J.M. Nn-kog2p: A novel grapheme-to-phoneme model for korean language. In Proceedings of the ICASSP 2021–2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Toronto, ON, Canada, 6–11 June 2021; pp. 7783–7787.
33. Řezáčková, M.; Švec, J.; Tihelka, D. T5g2p: Using text-to-text transfer transformer for grapheme-to-phoneme conversion. In Proceedings of the INTERSPEECH, Brno, Czechia, 30 August–3 September 2021; pp. 6–10.
34. Řezáčková, M.; Frémund, A.; Švec, J.; Matoušek, J. T5G2P: Multilingual Grapheme-to-Phoneme Conversion with Text-to-Text Transfer Transformer. In Proceedings of the Asian Conference on Pattern Recognition, Springer, Kitakyushu, Japan, 5–8 November 2023; pp. 336–345.
35. Řezáčková, M.; Tihelka, D.; Matoušek, J. T5g2p: Text-to-text transfer transformer based grapheme-to-phoneme conversion. *IEEE/ACM Trans. Audio, Speech Lang. Process.* **2024**, *32*, 3466–3476. [[CrossRef](#)]
36. Dong, L.; Guo, Z.Q.; Tan, C.H.; Hu, Y.J.; Jiang, Y.; Ling, Z.H. Neural grapheme-to-phoneme conversion with pre-trained grapheme models. In Proceedings of the ICASSP 2022–2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Singapore, 22–27 May 2022; pp. 6202–6206.
37. Tsvetkov, Y.; Sitaram, S.; Faruqui, M.; Lample, G.; Littell, P.; Mortensen, D.; Black, A.W.; Levin, L.; Dyer, C. Polyglot neural language models: A case study in cross-lingual phonetic representation learning. *arXiv* **2016**, arXiv:1605.03832.
38. Milde, B.; Schmidt, C.; Köhler, J. Multitask Sequence-to-Sequence Models for Grapheme-to-Phoneme Conversion. In Proceedings of the INTERSPEECH, Stockholm, Sweden, 20–24 August 2017; pp. 2536–2540.
39. Pritzen, J.; Gref, M.; Zühlke, D.; Schmidt, C. Multitask learning for grapheme-to-phoneme conversion of anglicisms in German speech recognition. *arXiv* **2021**, arXiv:2105.12708.
40. Route, J.; Hillis, S.; Etinger, I.C.; Zhang, H.; Black, A.W. Multimodal, multilingual grapheme-to-phoneme conversion for low-resource languages. In Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo 2019), Hong Kong, China, 3 November 2019; pp. 192–201.
41. Vesik, K.; Abdul-Mageed, M.; Silfverberg, M. One model to pronounce them all: Multilingual grapheme-to-phoneme conversion with a transformer ensemble. *arXiv* **2020**, arXiv:2006.13343.
42. Sokolov, A.; Rohlin, T.; Rastrow, A. Neural machine translation for multilingual grapheme-to-phoneme conversion. *arXiv* **2020**, arXiv:2006.14194.
43. Engelhart, E.; Elyasi, M.; Bharaj, G. Grapheme-to-phoneme transformer model for transfer learning dialects. *arXiv* **2021**, arXiv:2104.04091.
44. Ni, J.; Shiga, Y.; Kawai, H. Multilingual Grapheme-to-Phoneme Conversion with Global Character Vectors. In Proceedings of the Interspeech, Hyderabad, India, 2–6 September 2018; pp. 2823–2827.
45. Deri, A.; Knight, K. Grapheme-to-phoneme models for (almost) any language. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, Berlin, Germany, 7–12 August 2016; pp. 399–408.
46. Peters, B.; Dehdari, J.; van Genabith, J. Massively multilingual neural grapheme-to-phoneme conversion. *arXiv* **2017**, arXiv:1708.01464.
47. Lee, J.L.; Ashby, L.F.; Garza, M.E.; Lee-Sikka, Y.; Miller, S.; Wong, A.; McCarthy, A.D.; Gorman, K. Massively multilingual pronunciation modeling with WikiPron. In Proceedings of the Twelfth Language Resources and Evaluation Conference, Marseille, France, 11–16 May 2020; pp. 4223–4228.
48. Gorman, K.; Ashby, L.F.; Goyzueta, A.; McCarthy, A.D.; Wu, S.; You, D. 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, Online, 10 July 2020; pp. 40–50.
49. Mortensen, D.R.; Dalmia, S.; Littell, P. Epitran: Precision G2P for many languages. In Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), Miyazaki, Japan, 7–12 May 2018; pp. 2710–2714.
50. Yu, M.; Nguyen, H.D.; Sokolov, A.; Lepird, J.; Sathyendra, K.M.; Choudhary, S.; Mouchtaris, A.; Kunzmann, S. Multilingual grapheme-to-phoneme conversion with byte representation. In Proceedings of the ICASSP 2020–2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Online, 4–8 May 2020; pp. 8234–8238.
51. Kim, H.Y.; Kim, J.H.; Kim, J.M. Fast Bilingual Grapheme-To-Phoneme Conversion. In Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Industry Track, Seattle, DC, USA, 10–15 July 2022; pp. 289–296.
52. Zhu, J.; Zhang, C.; Jurgens, D. ByT5 model for massively multilingual grapheme-to-phoneme conversion. *arXiv* **2022**, arXiv:2204.03067.
53. Glocker, K.; Herygers, A.; Georges, M. Allophant: Cross-lingual Phoneme Recognition with Articulatory Attributes. *arXiv* **2023**, arXiv:2306.04306.
54. Sejnowski, T.; Rosenberg, C. Connectionist Bench (Nettalk Corpus). *Uci Mach. Learn. Repos.* **1988**. [[CrossRef](#)]

55. Park, K.; Lee, S. g2pm: A neural grapheme-to-phoneme conversion package for mandarin chinese based on a new open benchmark dataset. *arXiv* **2020**, arXiv:2004.03136.
56. Veisi, H.; MohammadAmini, M.; Hosseini, H. Toward Kurdish language processing: Experiments in collecting and processing the AsoSoft text corpus. *Digit. Scholarsh. Humanit.* **2020**, *35*, 176–193. [[CrossRef](#)]
57. Rahmati, E.; Sameti, H. GE2PE: Persian End-to-End Grapheme-to-Phoneme Conversion. In Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2024, Miami, Florida, 12–16 November 2024; pp. 3426–3436.
58. Alghmadi, M. KACST arabic phonetic database. In Proceedings of the the Fifteenth International Congress of Phonetics Science, Barcelona, Spain, 3–9 August 2003; pp. 3109–3112.
59. Halpern, J. THE ROLE OF PHONETICS AND PHONETIC DATABASES IN JAPANESE SPEECH TECHNOLOGY. 2008. Available online: <https://api.semanticscholar.org/CorpusID:43967749> (accessed on 9 November 2024).
60. Kurian, C. Speech database and text corpora for Malayalam language automatic speech recognition technology. In Proceedings of the IEEE 2016 Conference of The Oriental Chapter of International Committee for Coordination and Standardization of Speech Databases and Assessment Techniques (O-COCOSDA), Bali, Indonesia, 26–28 October 2016; pp. 7–11.
61. Ipa-Dict. Monolingual Wordlists with Pronunciation Information in IPA. Available online: <https://github.com/open-dict-data/ipa-dict> (accessed on 20 September 2016).
62. Chen, S.F. Conditional and joint models for grapheme-to-phoneme conversion. In Proceedings of the INTERSPEECH, Geneva, Switzerland, 1–4 September 2003; pp. 2033–2036.
63. Galescu, L.; Allen, J.F. Pronunciation of proper names with a joint n-gram model for bi-directional grapheme-to-phoneme conversion. In Proceedings of the Seventh International Conference on Spoken Language Processing, Denver, CO, USA, 16–20 September 2002; pp. 109–112.
64. Zhao, C.; Wang, J.; Qu, X.; Wang, H.; Xiao, J. r-g2p: Evaluating and enhancing robustness of grapheme to phoneme conversion by controlled noise introducing and contextual information incorporation. In Proceedings of the 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Singapore, 23–27 May 2022; pp. 6197–6201.
65. Wang, Y.; Bao, F.; Zhang, H.; Gao, G. Joint alignment learning-attention based model for grapheme-to-phoneme conversion. In Proceedings of the ICASSP 2021–2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Toronto, ON, Canada, 6–11 June 2021; pp. 7788–7792.
66. Cao, D.; Zhao, Y.; Wu, L. Near-Optimal Active Learning for Multilingual Grapheme-to-Phoneme Conversion. *Appl. Sci.* **2023**, *13*, 9408. [[CrossRef](#)]
67. Qharabagh, M.F.; Dehghanian, Z.; Rabiee, H.R. LLM-Powered Grapheme-to-Phoneme Conversion: Benchmark and Case Study. *arXiv* **2024**, arXiv:2409.08554.
68. Han, D.; Cui, M.; Kang, J.; Wu, X.; Liu, X.; Meng, H. Improving Grapheme-to-Phoneme Conversion through In-Context Knowledge Retrieval with Large Language Models. *arXiv* **2024**, arXiv:2411.07563.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.