Bangla Grapheme to Phoneme Conversion Using Conditional Random Fields

Conference Paper · December 2017

DOI: 10.1109/ICCITECHN.2017.8281780

CITATIONS
2

READS
344

4 authors:

Shammur Absar Chowdhury
Qatar Computing Research Institute
46 PUBLICATIONS **225** CITATIONS
SEE PROFILE

Firoj Alam
Qatar Computing Research Institute
60 PUBLICATIONS **489** CITATIONS
SEE PROFILE

Naira Khan
Institute of Education and Research
12 PUBLICATIONS **30** CITATIONS
SEE PROFILE

Sheak Rashed Haider Noori
Daffodil International University
26 PUBLICATIONS **55** CITATIONS
SEE PROFILE

2017 20th International Conference of Computer and Information Technology (ICCIT), 22-24 December, 2017

Bangla Grapheme to Phoneme Conversion Using Conditional Random Fields

Shammur Absar Chowdhury
University of Trento, Italy
shammur.chowdhury@unitn.it

Firoj Alam
QCRI, Qatar
fialam@hbku.edu.qa

Naira Khan
Dhaka University, Bangladesh
nairakhan@du.ac.bd

Sheak R. H. Noori
DIU, Bangladesh
drnoori@daffodilvarsity.edu.bd

***Abstract*—Integrated with handheld devices, toys, KIOSKs, and call centers, Text to Speech (TTS) and Speech Recognition (SR) have become widely used applications in everyday life. One of the core components of said applications is Grapheme to Phoneme (G2P) conversion. The task at hand is the mapping of the written form to the spoken form, i.e. mapping one sequence to another. In Natural Language Processing (NLP), it is typically referred to as a sequence to sequence labeling task. The task however, is a language dependent one and has primarily been implemented for English and similar resource-rich languages. In comparison, very little has been done for digitally under-resourced languages such as Bangla (ethnonym: Bangla; exonym: Bengali). The current state-of-the-art Bangla Grapheme to Phoneme conversion is limited to rule-based and lexicon based approaches, the development of which requires a significant contribution of linguistic experts. In this paper, we propose a data-driven machine learning approach for Bangla G2P conversion. We evaluate the existing rule based approaches and design a machine learning model using Conditional Random Fields (CRFs). To train the machine learning models we have only used character level contextual features due to the fact that extracting hand crafted features requires specialized knowledge. We have evaluated the systems using two publicly available datasets. We have obtained promising results with a phoneme error rate of 1.51% and 14.88% for CRBLP and Google pronunciation lexicons, respectively.**

***Keywords*—Bangla, Conditional Random Fields, Pronunciation Generation, Grapheme to Phoneme (G2P)**

I. Introduction

Although our daily interactions are primarily dominated by speech or spoken conversation as the primary mode of communication, written communication also occupies a significant space in the communication sphere of human civilization. As such it is necessary to access written speech even if one is visually impaired. Therefore, it is significantly vital for the visually impaired to have access to synthesized speech of a written text. For machine understanding and generation, specifically for speech synthesis, i.e., Text to Speech (TTS)) and Automatic Speech Recognition (ASR) systems, one important step is to provide a mapping between orthographic and phonetic representations. For said mapping task, we need to infer one from the other, i.e., from orthographic to phonetic form and vice-versa. The notion of G2P is the that it takes a word (i.e., orthographic representation) e.g., DUKE, and generates a phonemic or phonetic representation, e.g., /d uw k/. An example in Bangla is as follows: আদেশ /a d e sh/ (order). The G2P system examines the grapheme sequence and utilizes different rules/techniques to generate a phoneme sequence. In relevant literature, it is also referred to as a letter to sound mapping [1].

In the early days of computational G2P research, a typical approach was to use a digitised pronunciation lexicon[1], manually developed by lexicographers and linguists. For example, a publicly available pronunciation lexicon for English is the CMU Dictionary [2][2], and for Bangla it is the CRBLP Pronunciation Lexicon [3][3] and Google's Bangla pronunciation lexicon [4]. The limitation of a lexicon-based approach is that an automated system is not able to provide a pronunciation of an unknown word. Another limitation is that it is memory intensive to load a large list of a lexical items, especially for hand-held devices.

Another early approach, based on implementing a deterministic system, utilised pronunciation rules devised by linguists. Some earlier work on the rule-based approaches for English can be found in [5], [6], [7], [8], [9]. For Bangla, the research is sparse and one of the seminal studies can be found in [10], later extended in the study of Alam et al. [3]. Other relevant research includes [11], [12].

Data-driven statistical machine learning approaches are not new, however, research efforts in said approach is sparse. The data-driven approach requires a lexicon containing an exhaustive list of the pronunciation of the words in order to train a machine learning model. For English, the earliest work is done by Sejnowski et al. [13], [14] using a feed-forward neural network, comprising one input, a hidden and an output layer. The alternative machine-learning based approach includes the use of decision trees [15]. A comparative study has been done in [16] using several algorithms. We discuss more details about different approaches in Section II.

Compared to the research on English, the only efforts for Bangla that we are aware of was done by [17], in which they trained a machine learning model using $37K$ words. The model was developed to facilitate a transcriber and the reported accuracy is 81.5%. In this study, we explore a CRFs based machine learning approach for Bangla G2P conversion. Our contributions include:

1) we provide a systematic comparison with existing rule based approaches, such as that in [10] and [3], using publicly available pronunciation lexicons like CRBLP [3].

[1]A correspondences between orthography and its pronunciation of a word
[2]https://github.com/cmusphinx/cmudict
[3]Available as part of a Bangla Text to Speech system: https://github.com/firojalam/Katha-Bangla-TTS

978-1-5386-1150-0/17/$31.00

2) train and evaluate them using CRFs models by exploiting the same datasets, and
3) make the experimental resources publicly available on Github[4] in order to ensure replicability and to enable future research.

The structure of the current paper is as follows: In Section II, we provide a brief overview of related work. We discuss the details of the dataset in Section III and present the methodology of said study in Section IV. We present experimental details in Section V, and discuss the results in Section VI, with concluding remarks in Section VII.

II. Related Work

In this Section, we first discuss the current state-of-the-art G2P for different languages. We then review related work for Bangla G2P completed till date, including lexicon and rule-based approaches, and discuss the limitations.

There has been extensive work on grapheme to phoneme conversion - from rule based approaches to the current state-of-the-art deep-learning based approach. Most of the work pertains to English. The study of Chen [18] proposed several machine learning models such as a conditional maximum entropy model, a joint maximum entropy n-gram model and a joint maximum entropy n-gram model with syllabification. The author reports results using Phoneme-Error-Rate (PER)[5] and Word-Error-Rate (WER)[6] metrics. For the CMU pronunciation dictionary the reported best PER is 1.4% and WER is 8.4%. Mana et al. [20] proposed a Classification-and-Regression-Tree (CART) for developing a G2P system for three different languages i.e., British and American English, French, and Brazilian Portuguese. The study of Thu et al. [16] provides a good comparison using different machine learning methods for G2P conversion of the Burmese language - another under-resourced language. They experimented and evaluated seven different algorithms using a small amount of label dataset (i.e., pronunciation lexicon). These include Support Vector Machines (SVMs), Conditional Random Fields (CRFs), Joint-Sequence Models (JSMs), Weighted Finite-state Transducers (WFST), and Recurrent Neural Networks (RNN). Their findings suggest that CRFs is one of the best performing classifiers.

The recent development of deep learning models also shows promise for G2P tasks. Yao and Zweig [21] used bi-directional Long Short Term Memory neural networks (LSTMs) and report a PER rate of 5.45% on the CMU pronunciation lexicon. Their performance is better than the joint-sequence model by [22].

An extensive cross-language study has been conducted by Kim and Snyder [23] for latin alphabets where they experimented and evaluated their system using 107 languages. Their study with an undirected graphical model provides an F1 measure of 88%. Deri and Knight [24] has conducted another cross language G2P study using an adaptive approach where they explored 229 languages. For Bangla their reported WER is 66.2%.

The linguistic study of Bangla phonetics is dated back to 1921 by Chatterji [25]. Other notable studies include the work by Ferguson and Chowdhury [26], Haque [27], Imtiaz [28], Hai [29].

Mosaddeque et al. [10] adopted the linguistic rules proposed in the Bangla Academy pronunciation dictionary [30]. Their implemented G2P system achieves an accuracy of 97.01% on a seen corpus[7] of 736 words. On an unseen corpus[8] of 8399 words it achieves an accuracy of 81.95%. An extended version of this work, consists of 3880 rules as reported in [3], achieves an accuracy of 89.48%, and has been evaluated on a different corpus. The evaluation is not exactly comparable, as the later system has not been evaluated on the same corpus. Therefore, it is difficult to say that the later system significantly improved the performance compared to the former. Our study will address the above issue by comparing both systems on the same corpus. More details can be found in IV-A.

To our knowledge, the existing rule-based Bangla G2P systems are as follows: Basu et al. [12] proposed a rule based system, which takes into account three types of information, i.e. orthographic information with their preceding and succeeding context, Parts-of-Speech (POS), and the semantic content. In addition, they consider an exception dictionary. Their system consists of only 21 rules with which it achieves an accuracy of 91.48% on 9294 words from 1000 sentences. The work of Gosh et al. [11] proposed a heuristic based approach containing orthographic, POS and contextual information. Their system achieves an accuracy of 70%, which has been evaluated on a set of randomly selected 755 words.

The first publicly available Bangla pronunciation lexicon contains $\sim 93K$ entries, as a part of the Bangla Text to Speech system [3]. The system also contains syllabified information, which can be useful for an automatic syllable boundary detection problem. A similar lexicon has been developed by Google and made publicly available [4], [17] containing $\sim 65K$ entries. These are the resources upon which our research is based.

One limitation of the existing rule based efforts for Bangla is that they are hardly comparable due to the lack of 1) non-sharable resources such as data and/or the system, 2) not maintaining the same evaluation condition or performance metrics. In this study, we try to overcome those limitations by maintaining widely used performance metrics such as Phoneme Error Rate (PER) and Word Error Rate (WER). Moreover, we make the experimental resources publicly available for future research.

III. Data

We discussed earlier that our study is based on two publicly available pronunciation lexicons. For the training and evaluation of the systems we split them into three different sets i.e. training, development and test set with a proportion of 70%, 15% and 15% respectively. Such a split is typical in any machine learning task. The training set is used to train the model, the development set is used to optimize the learning parameters during the training process based on the PER, and

[4]https://github.com/cogniinsight/Bangla-G2P/

[5]The minimal number of insertions, deletions, and substitutions between a reference and predicted (hypothesis) pronunciation. Typically computed using edit distance algorithm such as the Levenshtein algorithm [19]

[6]How often the pronunciation of a word is not completely correct.

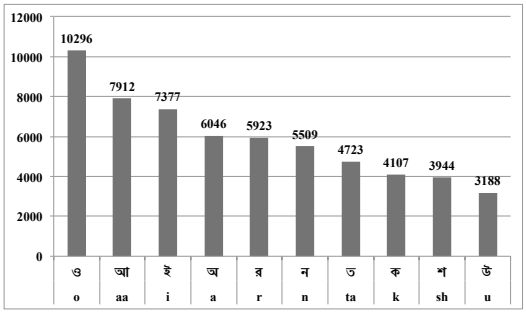
[7]A seen corpus is the one that has been used to investigate patterns and implement and evaluate pronunciation rules.

[8]An unseen corpus is a corpus that has been randomly chosen to evaluate pronunciation rules.

finally the test set is used for evaluation. In Table II, we provide a distribution of these datasets.

CRBLP lexicon: The first version of the CRBLP lexicon consists of $\sim 93K$ entries. Each entry contains information about the syllabic boundary, syllabic information (e.g., CVC pattern), Bangla pronunciation and the IPA transcription. An example entry is presented in Table I. For our study, we used a subset of this data consisting of $\sim 77K$ entries. The phoneme set includes 30 consonants, 14 monophthong vowels (oral and nasal) and 21 diphthongs [31], [32]. The top ten most frequent phonemes in the lexicon are shown in Figure 1.

Figure 1: Top ten most frequent phonemes in CRBLP lexicon. Phonemes are represented in ASCII form.



Google lexicon: The Google lexicon contains three different pieces of information i.e. orthography, IPA transcription with syllable information, and an optional disambiguating label. It consists of $65K$ entries and the number of phonemes is 39. The top ten most frequent phonemes in the lexicon is shown in Figure 2. An example entry is shown in Table I. Compared to the CRBLP lexicon it lacks the Bangla pronunciation of an orthographic form, which limits the alignment task discussed below. One important reason of having a reduced phone set is that they mostly ignored diphthongs and nasal vowels.

Figure 2: Top ten most frequent phonemes in Google lexicon. Phonemes are represented in ASCII form.

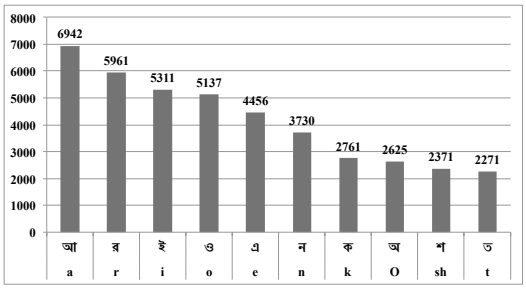


Table I: An example of an entry in the lexicons

Lexicon	Orthography	Bangla Pronunciation	Transcription
Google	আমার		a . m a r
CRBLP	আমার	আ.মার্	a . m a r

Several interesting points have been observed upon analyzing the frequency of phonemes in two lexicons. The most frequent top ten entries in these lexicons has 90% matches. We see the mismatch between phoneme ঊ /u/ (appears in CRBLP lexicon not in Google) and এ/ে /e/ (appears in Google lexicon, not CRBLP). This analysis provides evidence that these are the most frequent phonemes for Bangla irrespective of their order or position in the frequency list.

Note that there is a discrepancy between the number of phonemes in two lexicons. In this study, we are not focusing on resolving said issue, and it is beyond the scope of the current paper.

Alignment:

Like many other languages, for Bangla, there can also be a lack of direct correspondence between a grapheme and a phoneme, which makes the task difficult. For example, ক্ষ corresponds to phonemes /k/ and /k^h/ in gemination. Also, there is an issue with homographs. For example, if a word starts with অ /ɔ/ followed by ক্ষ then the অ /ɔ/ is pronounced as ও /o/. Such challenges have also been reported in [17].

To design a sequence-to-sequence model a proper alignment between a grapheme and a phoneme is necessary for each word. Among the two lexicons, none of them has a direct mapping. Therefore, in our study, we have conducted experiments using two different alignment procedures:

- Weak alignment: This alignment first assumes that the length of the grapheme and phoneme sequence is the same and there is a direct mapping. An example of an alignment is given in Table III. However, if there is a discrepancy in lengths between the two, then we shift them at the left and a dummy symbol is added at the right. This procedure maps many graphemes to a dummy symbol (e.g., #) and vice versa. Another misalignment occurs for halant (্), -- Bengali sign virama U+09CD.
- Strong alignment: For this alignment, we exploit syllable boundary and a apply weak alignment procedure for each syllable. Although this procedure introduces some misalignments, it improves the alignment significantly, as we will see from classification experiments.

Our alignment approaches are simple and easy to implement, given little to no alignment information. In the literature, there are other techniques for alignment such as Dynamic Time Warping (DTW) [1]. It assigns a cost to each phoneme to grapheme alignment. Good alignments are given low costs and bad alignments are given highest costs. The algorithm then searches all possible alignments and picks the one with the lowest total cost.

Table II: Distribution of the datasets with training, development and test data split.

Data	Train (70%)	Dev (15%)	Test (15%)	Total
CRBLP	54003	11556	11557	77116
Google	45527	9756	9756	65039

For the current study, we experimented both alignment approaches using the CRBLP lexicon and applied a weak alignment for Google lexicon.

IV. Methodology

In this Section, we first discuss our evaluation of the existing rule based systems and then discuss the CRFs model.

A. Evaluation of Existing Rule Based Systems

For evaluating the existing rule based systems, we adopted the G2P systems by Mosaddeque et al. [10] and and Alam et al. [3]. We evaluated these two systems using both the development and test sets of the CRBLP lexicon. The rule based systems do not require any alignment procedure, since the phoneme sequence is generated using linguistic rules from the grapheme sequence. We could not evaluate these systems using the Google lexicon due to the mismatch of the phonetic transcription.

B. CRFs Model

We designed our sequence classification model using CRFs [33]. It is a popular probabilistic graphical model that can exploit the dependency structure. It has been widely used in Natural Language Processing (NLP), Computer Vision and bioinformatics. In our case, the dependency is basically a contextual dependence (i.e., previous and following context). For a given grapheme sequence $\boldsymbol{x} = \{x_0, x_1, ...x_i, ...x_T\}$ it tries to find the best phoneme sequence $\boldsymbol{y} = \{y_0, y_1, ...y_i, ...y_N\}$ by modeling conditional distribution $P(\boldsymbol{y}|\boldsymbol{x})$. Each x_i is basically a vector containing information about the grapheme such as its identity, previous graphemes and so on, and each y_j represents a phoneme. The best phoneme sequence can be obtained by computing the maximum probability as defined by the eq. 1.

$$\boldsymbol{y}* = argmax_y P(\boldsymbol{y}|\boldsymbol{x}) \tag{1}$$

The probability of the phoneme sequence is modeled by the learned parameters and feature functions as shown in eq. 2

$$P(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\lambda}) = \frac{1}{Z(\boldsymbol{x})} exp\left(\sum_{i=1}^{T}\sum_{j}\lambda_j f_j(y_{i-1}, y_i, x_i)\right) \tag{2}$$

where $f_j(y_{i-1}, y_i, x, i)$ is the feature function and $\boldsymbol{\lambda}$ is the learned parameter. Here, for the sake of simplicity, the feature function considers the current (y_i) and one previous (y_{i-1}) phoneme, and the current grapheme x_i. The observation dependent normalization $Z(\boldsymbol{x})$ is defined by the equation 3. It ensures the distribution of P sums to 1. The feature function can be designed many different ways. One example is the boolean representation, i.e., presence or absence of a characteristic for a grapheme x_i.

$$Z(\boldsymbol{x}) = \sum_{y\in\boldsymbol{y}}\sum_{i=1}^{T}\sum_{j}\lambda_j f_j(y_{i-1}, y_i, x_i) \tag{3}$$

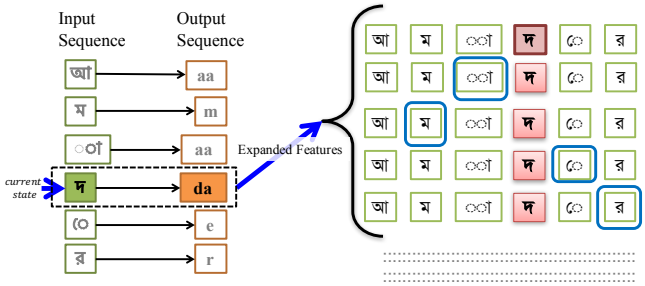
The parameter $\boldsymbol{\lambda}$ is learned using the maximum likelihood estimation from the training data, $D = [(\boldsymbol{x}^1, \boldsymbol{y}^1), (\boldsymbol{x}^2, \boldsymbol{y}^2), ..., (\boldsymbol{x}^m, \boldsymbol{y}^m)]$, that we provide during training. The mathematical details of the parameter estimation is omitted here for simplicity, the details can be found here [33], [34].

V. Experiments

A. Training

We trained CRF models using both weak and strong alignments of the CRBLP-lexicon data along with the Google dataset. For training, we designed the feature vector for each grapheme by exploiting the grapheme itself along with its contextual/surrounding grapheme. The context is defined by n-grams -- unigram and bigram. An example of the input and the output sequence along with some extended features containing n-gram features presented in Figure 3. In addition to the context window of the input sequences, a combination of the current output token (i.e., দ/da/) and previous output token (i.e., া/aa/) is automatically generated. For CRBLP-weak, CRBLP-strong alignment settings, a total of $\approx 27K$ and $\approx 37K$ features are extracted respectively, whereas for the Google model training $\approx 20K$ features are extracted in total.

Figure 3: An example of input and output sequences along with extended feature representation. Phonemes are represented in ASCII form. → and red box represents the current grapheme and the blue box represents the preceding and succeeding contexts.



B. Evaluation

To evaluate the performance of the system, we computed PER and WER. The formula for PER and WER is shown in the equation 4. The evaluation requires a further alignment procedure. To have a clear understanding, in Table III and IV we provide an expected alignment and the predicted alignment of a classifer, for the word গৃহপালিত (Domesticated) /g r i . h o . p a . l i . t̪ o/.

To align the phoneme sequence of the classifier prediction with reference sequence, a typical approach is to use a dynamic programming algorithm [35]. It requires creating a distance matrix with one column for each phoneme in the phoneme sequence and one row for each grapheme in the grapheme sequence. From the distance matrix, PER is computed based on the edit operations such as insertion **I**, deletion **D**, substitution **S**. The computation of WER is based on PER, which takes into account of the words that have at least one edit operation (error).

$$\begin{aligned} PER &= \frac{100 * (I + S + D)}{Total\ Phonemes} \\ WER &= \frac{100 * (\#\ of\ Words\ with\ phoneme\ error)}{Total\ Words} \end{aligned} \tag{4}$$

Table III: Example of an expected mapping. For phonetic representation here we used ASCII representation instead of IPA for easier machine processing.

Example of mapping for the word গৃহপালিত										
Graphemes	গ	ৃ		হ	প	া	ল	ি	ত	
Phonemes	g	r	i	ho	p	aa	l	i	ta	o

Table IV: Example of reference and hypothesis/prediction and their mapping/alignment. * represent the system deleted those characters and ** represent it inserted those characters by mistake. Here we have 2 insertions, 2 deletions, and no substitutions. Error = 36.36%

A mapping between reference and hypothesis for the word গৃহপালিত													
Ref	g	r	i	h	o	p	aa	l	i	**	**	ta	o
Hyp	g	r	i	h	*	*	aa	l	i	ta	o	ta	o

VI. Results, Analysis and Future Study

Results: In Table V and VI, we present the performance of two different datasets. The results in Table V demonstrate the performances of two different rule based systems and two of our alignment approaches. The evaluation results on the same dataset show that there is a significant performance difference in rule-based vs machine-learning based approach. Both rule-based approaches perform quite similarly. There is also a large difference in performances of the results between two alignment procedures. This shows us evidence that it is necessary to investigate the possible alignment procedures in order to find a suitable one. Our CRFs results with strong alignment are quite similar to the current state-of-the-art results for English (See the results of the CMU pronunciation lexicon in [21]).

Table V: PER and WER, in percentage (%), of two different datasets with development and test set split. Lower PER and WER is better.

Approach	**Data**	**PER**	**WER**
Rule (Mosaddeque et al., 2006)	**Dev**	52.39	98.75
	Test	52.55	98.88
Rule (Alam et al., 2011)	**Dev**	52.54	98.86
	Test	52.39	98.75
CRFs: Weak Alignment	**Dev**	22.04	66.10
	Test	21.99	66.12
CRFs: Strong Alignment	**Dev**	1.41	8.97
	Test	1.51	9.69

The results in Table VI are obtained using the Google dataset. For this dataset, we were able to train and evaluate using the weak alignment procedure, due to the lack of syllabic information and the mismatch between the phoneme set. Even if the results of two datasets are not exactly comparable, however, if we compare the results of the weak alignment approach of these two datasets, we can infer that the weak alignment performs better with the Google dataset. The results using the Google dataset is a first step in using it in future research and will serve as a point of comparison.

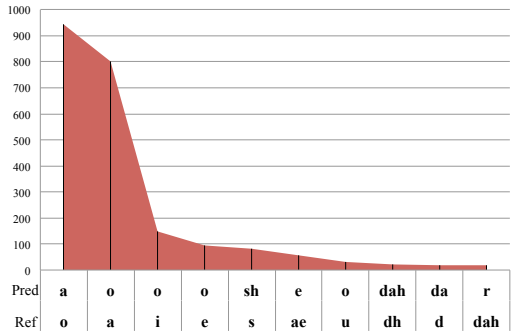
Error Analysis: To have an in-depth understanding of the phoneme mismatches we created a confusion matrix of the reference and predicted the phoneme from the results of the three trained models. For the CRBLP dataset, we have

Table VI: PER and WER of Google dataset with development and test set split.

Data	**PER**	**WER**
Dev	15.33	46.32
Test	14.88	46.61

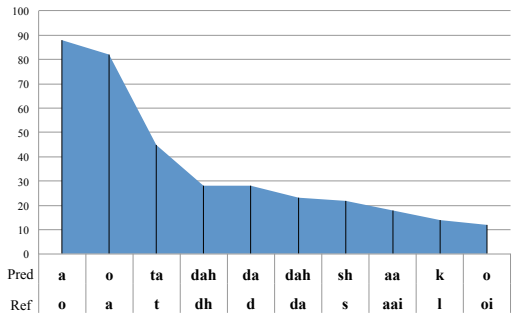
observed that in case of weak-alignment, as shown in Figure 4, most confusion occurs between the phoneme **/o/** and **/a/**. The phoneme **/o/** has also been recognized as /aa/ and /oi/.

Figure 4: Top ten confused phonemes in classification using weak alignment approach of the CRBLP lexicon. Ref. and Pred. represent the reference and hypothesis/prediction. Phonemes are represented in ASCII form.



Similarly, in the case of strong alignment we observed that the same subset of phonemes such as /o/-/a/, /a/-/o/, /s/-/sh/, /dh/-/dah/ among others, are most frequently confused with phoneme pairs. But with strong alignment, we observed that the tendency to confuse these pairs are significantly decreased, as shown in Figure 5.

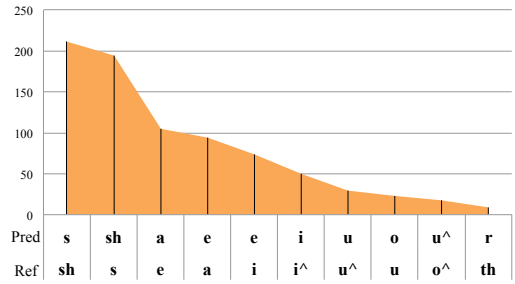
Figure 5: Top ten confused phonemes in classification using strong alignment approach of the CRBLP lexicon. Ref. and Pred represents the reference and hypothesis/prediction. Phonemes are represented in ASCII form.



As for the Google dataset, we observed that the most frequently confused pairs of phonemes are /s/-/sh/, /a/-/e/, /e/-/i/ among others, as presented in Figure 6. Similar findings have also been reported in [17].

Future Research Directions: There are many research directions that can be adopted to address the current problem. First of all, we need to develop a common test data from both lexicons to evaluate the system's performance, which can be

Figure 6: Top ten confused phonemes in classification using weak alignment approach of the Google lexicon. Ref. and Pred represents the reference and hypothesis/prediction. Phonemes are represented in ASCII form.



trained using different datasets. One possible research problem would be finding common entries in both lexicons and to make that a test set. The second problem that we can identify is the different phoneme representations. Future research could also focus on making a common representation. Once this process is done both datasets can be combined to design a better machine learning model. As we have seen better grapheme to phoneme mapping can provide improved performance, which is obtained using syllable information, therefore, an interesting research problem would be syllable boundary detection.

From a machine learning point of view, one interesting research problem would be exploiting deep neural networks such as LSTM and character embedding (i.e., character level distributed representation), and it might provide much better results.

VII. Conclusions

In this paper, we presented our study of Bangla grapheme to phoneme conversion. We evaluated the existing rule based approaches and compared that with our proposed CRFs based models. We explored two different alignment approaches. We observed that better alignment provide improved performance. For training and evaluating the proposed system, we used two different publicly available datasets. For future research, we plan to make the experimental data and resources accessible through Github. We obtained significantly better results compared to the rule based system results. We have also discussed limitations and provided future research directions.

References

[1] P. Taylor, Text-to-speech synthesis. Cambridge university press, 2009.

[2] R. Weide, ``The cmu pronunciation dictionary, release 0.6," Carnegie Mellon University, 1998.

[3] F. Alam, S. M. Habib, and M. Khan, ``Bangla text to speech using festival," HLTD 2011, 2011.

[4] Google. Bangla phonology and lexicon.

[5] W. Ainsworth, ``A system for converting english text into speech," IEEE Transactions on audio and electroacoustics, vol. 21, no. 3, 1973.

[6] M. D. McIlroy, ``Synthetic english speech by rule," The Journal of the Acoustical Society of America, vol. 55, no. S1, pp. S55--S56, 1974.

[7] H. Elovitz, R. Johnson, A. McHugh, and J. Shore, ``Letter-to-sound rules for automatic translation of english text to phonetics," IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. 24, no. 6, pp. 446--459, 1976.

[8] J. Bachenko and E. Fitzpatrick, ``A computational grammar of discourse-neutral prosodic phrasing in english," Computational linguistics, vol. 16, no. 3, pp. 155--170, 1990.

[9] S. Hunnicutt, ``Phonological rules for a text-to-speech system," American Journal of Computational Linguistics, pp. 1--72, 1976.

[10] A. B. Mosaddeque, N. UzZaman, and M. Khan, ``Rule based automated pronunciation generator," 2006.

[11] K. Ghosh, R. V. Reddy, N. Narendra, S. Maity, S. Koolagudi, and K. Rao, ``Grapheme to phoneme conversion in bengali for festival based tts framework," in Proc. of ICON. Macmillan Publishers, 2010.

[12] J. Basu, T. Basu, M. Mitra, and S. K. D. Mandal, ``Grapheme to phoneme (g2p) conversion for bangla," in Proc. of COCOSDA (Speech Database and Assessments). IEEE, 2009, pp. 66--71.

[13] T. J. Sejnowski and C. R. Rosenberg, ``Parallel networks that learn to pronounce english text," Complex systems, vol. 1, no. 1, pp. 145--168, 1987.

[14] T. J. Sejnowski and C. R. Rosenberg, NETtalk: A parallel network that learns to read aloud. MIT Press, 1988.

[15] V. Pagel, K. Lenzo, and A. Black, ``Letter to sound rules for accented lexicon compression," arXiv preprint cmp-lg/9808010, 1998.

[16] Y. K. Thuλ, W. P. Pa, Y. Sagisaka, and N. Iwahashiλ, ``Comparison of grapheme--to--phoneme conversion methods on a myanmar pronunciation dictionary," WSSANLP 2016, p. 11, 2016.

[17] A. Gutkin, L. Ha, M. Jansche, K. Pipatsrisawat, and R. Sproat, ``Tts for low resource languages: A bangla synthesizer." in LREC, 2016.

[18] S. F. Chen et al., ``Conditional and joint models for grapheme-to-phoneme conversion." in INTERSPEECH, 2003.

[19] V. I. Levenshtein, ``Binary codes capable of correcting deletions, insertions, and reversals," in Soviet physics doklady, vol. 10, no. 8, 1966, pp. 707--710.

[20] F. Mana, P. Massimino, and A. Pacchiotti, ``Using machine learning techniques for grapheme to phoneme transcription," in Seventh European Conference on Speech Communication and Technology, 2001.

[21] K. Yao and G. Zweig, ``Sequence-to-sequence neural net models for grapheme-to-phoneme conversion," arXiv preprint arXiv:1506.00196, 2015.

[22] M. Bisani and H. Ney, ``Joint-sequence models for grapheme-to-phoneme conversion," Speech communication, vol. 50, no. 5, pp. 434--451, 2008.

[23] Y.-B. Kim and B. Snyder, ``Universal grapheme-to-phoneme prediction over latin alphabets," in Proc. of EMNLP. ACL, 2012, pp. 332--343.

[24] A. Deri and K. Knight, ``Grapheme-to-phoneme models for (almost) any language." in ACL (1), 2016.

[25] S. K. Chatterji, ``Bengali phonetics," Bulletin of the School of Oriental and African Studies, vol. 2, no. 1, pp. 1--25, 1921.

[26] C. A. Ferguson and M. Chowdhury, ``The phonemes of bengali," Language, vol. 36, no. 1, pp. 22--59, 1960.

[27] D. Huq, Bhasha Bigganer Katha (Facts about Linguistics). Mowla Brothers, Dhaka, 2002.

[28] Z. I. Ali, ``Dhanibijnaner bhumika (introduction to linguistics)," 2001.

[29] A. Hai, ``Dhvani vijnan o bangla dhvani-tattwa," 2007.

[30] Bangla Uchcharon Obhidhan (Bangla Pronunciation Dictionary). Bangla Academy.

[31] F. Alam, S. Habib, and M. Khan, ``Acoustic analysis of bangla vowel inventory," 2008.

[32] F. Alam, S. M. Habib, and M. Khan, ``Research report on acoustic analysis of bangla vowel inventory," Center for Research on Bangla Language Processing, BRAC University, 2008.

[33] J. Lafferty, A. McCallum, and F. Pereira, ``Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in Proc. of 8th ICML, vol. 1, 2001, pp. 282--289.

[34] C. Sutton, A. McCallum et al., ``An introduction to conditional random fields," Foundations and Trends® in Machine Learning, vol. 4, no. 4, pp. 267--373, 2012.

[35] J. H. Martin and D. Jurafsky, ``Speech and language processing," International Edition, vol. 710, 2000.