# ACOUSTIC UNIT DISCOVERY AND PRONUNCIATION GENERATION FROM A GRAPHEME-BASED LEXICON

*William Hartmann, Anindya Roy, Lori Lamel, and Jean-Luc Gauvain*

Spoken Language Processing Group, LIMSI-CNRS
91403 Orsay, France
{hartmann, roy, lamel, gauvain}@limsi.fr

## ABSTRACT

We present a framework for discovering acoustic units and generating an associated pronunciation lexicon from an initial grapheme-based recognition system. Our approach consists of two distinct contributions. First, context-dependent grapheme models are clustered using a spectral clustering approach to create a set of phone-like acoustic units. Next, we transform the pronunciation lexicon using a statistical machine translation-based approach. Pronunciation hypotheses generated from a decoding of the training set are used to create a phrase-based translation table. We propose a novel method for scoring the phrase-based rules that significantly improves the output of the transformation process. Results on an English language dataset demonstrate the combined methods provide a 13% relative reduction in word error rate compared to a baseline grapheme-based system. Our approach could potentially be applied to low-resource languages without existing lexicons, such as in the Babel project.

***Index Terms—*** acoustic unit discovery, automatic speech recognition, grapheme-based speech recognition, pronunciation learning

## 1. INTRODUCTION

While the majority of the components of an automatic speech recognizer are learned from both labeled and unlabeled data, the lexicon is still largely handmade by experts. The use of these lexicons inherently limits a system to previously defined representations. This is especially problematic for languages that do not have an expert-defined lexicon. With the recent interest in low-resource languages [1, 2], methods for automatically learning lexicons are required. We propose a method for both discovering acoustic units and building the pronunciation lexicon. To simplify the task, our approach relies on the assumption that a grapheme-based recognizer can be built that obtains reasonable performance. While performance may not be as strong as with a handcrafted phonetic dictionary, grapheme-based systems have been shown to work in a variety of languages [3]. Our work consists of contributions to both acoustic-unit discovery and pronunciation generation.

Previous work in acoustic unit discovery has mainly focused on the setting where no information is known about the language [1, 2, 4, 5]. Jansen and Church proposed a spectral clustering-based approach to learning acoustic units [1]. Since the clustering is based on HMM states, the resulting units correspond more to subphonetic units than actual phones. Lee and Glass avoid this issue by jointly learning acoustic units consisting of several states through a nonparametric Bayesian model [2]. Both approaches are able to learn models from unlabeled data, but neither approach addresses building a pronunciation dictionary or a complete ASR system that can be measured by word error rate (WER). Bacchiani and Ostendorf [5] used a joint modeling approach that produced a complete ASR system, but was also limited to single state acoustic units and could not build pronunciations for words not seen during training.

Pronunciation generation can be stated as the task of defining an acoustic representation for a word; this can be either an alternate representation for a known word or a new representation for a previously unseen word. Many approaches focus on utilizing what is known about the word—its lexical representation—to produce possible pronunciations. Earlier approaches used hand-derived rules to perform the conversion [6]. More recent approaches automatically learn a mapping from graphemes to phonemes [7]. Unlike in this work, these approaches assume the acoustic units are already known and defined. Automatic methods also assume a large training corpus. Though work has been done to reduce the amount of training data required [8], no training data exists for mapping graphemes to pronunciations using automatically discovered units. For deriving pronunciations from automatically discovered acoustic units, less work exists. Most approaches focus on the keyword spotting task and build the acoustic representation from an acoustic example of the word. Instead of representing a word by a string of acoustic units, a posteriorgram-like representation is used [1, 2]. The main limitation of these approaches is that building a standard ASR system from these representations is not possible.

Our approach represents a compromise between the task

of building an ASR system from zero resources [2] and the requirement of expert-defined acoustic units and pronunciation dictionaries. We assume the training data has been transcribed at the word-level and that some relationship exists between the orthography and pronunciation of the language. In Section 2 we describe our spectral clustering-based method for discovering acoustic units from context-dependent grapheme models. The method is similar to [1], but operates on whole HMMs instead of individual states. Initially, pronunciations are generated by directly mapping the original grapheme-based pronunciations to the new acoustic units. Section 3 presents the transformation process used to improve a previously defined lexicon. Given a set of pronunciation hypotheses for each word, phrase-based rules are learned. We present a method for scoring the phrase-based rules that leads to a reduction in WER compared to the original lexicon. The experimental setup and results are presented in Sections 4 and 5. In Section 6 we provide further analysis, and conclusions are presented in Section 7.

## 2. ACOUSTIC UNIT DISCOVERY

Our approach to acoustic unit discovery assumes that an initial context-dependent HMM-based system has been trained using a grapheme-based lexicon. As in other work [9], we assume the context-dependent grapheme models represent information similar to phonemes. We propose to cluster the context-dependent models into a set of acoustic units using spectral clustering [10].

Spectral clustering operates on a graph where the nodes are data points and the edges are similarities between the points. The specific implementations of spectral clustering algorithms can vary widely. We use the method described in Algorithm 1, originally proposed by Ng et al. [11], with two minor changes. Instead of using a fully connected similarity graph, we use a $k$-nearest neighbor graph, as recommended in [10], where $k$ is chosen to be the smallest value that still maintains a strongly connected graph. The final step in spectral clustering is the k-means algorithm. In [11], they claim that only a single clustering is necessary using their method for initialization, but we found better performance with using random restarts and selecting the clustering with the minimum within-class variance.

The spectral clustering approach hinges on the definition of similarity between data points—in our case, HMMs. Many approaches for measuring the similarity between HMMs exist, but most are computationally expensive. Since we needed to compute the similarity for several million pairs of HMMs, an efficient strategy was required. Our method is similar to the approach presented in [12], but we have adapted it for use with the left-to-right HMMs used in ASR—the original method requires the computation of the stationary distribution, which does not exist for the HMMs used in ASR.

**Algorithm 1** Spectral clustering algorithm adapted from Ng et al. [11]

**Input:** number $k$ clusters to create, similarity matrix $S \in \mathbb{R}^{n \times n}$.

Let $W$ be the $k$-nearest neighbor adjacency matrix built from $S$.

Let $D$ be a diagonal matrix where $d_{i,i} = \sum_{j=1}^{n} w_{i,j}$.

Compute graph laplacian: $L = I - D^{-1/2} W D^{-1/2}$.

Let $U \in \mathbb{R}^{nxk}$ be the matrix consisting of the first $k$ eigenvectors of $L$.

Unit normalize the rows of $U$.

For $i = 1, \ldots, n$, let $y_i \in \mathbb{R}^k$ be the $i$th row of $U$.

Using k-means, cluster the points $y_i$ into clusters $C_1, \ldots, C_k$.

**Output:** clusters $C_1, \ldots, C_k$.

The similarity between two HMMs is defined as

$$\text{HMM}_{\text{sim}}(\mathbf{h}, \mathbf{h}') = \sum_{a=1}^{A} \sum_{b=1}^{B} \frac{\alpha_{a,b}}{\text{CSD}(h_a, h'_b) + 1} \qquad (1)$$

where $\boldsymbol{h}$ and $\boldsymbol{h'}$ are HMMs with $A$ and $B$ number of states respectively. The occupancy matrix, $\boldsymbol{\alpha} \in \mathbb{R}^{A \times B}$, defines the probability of any state $h_a$ being occupied at the same time as state $h'_b$ over any sequence less than $N$ time steps; in this work, we use $N = 100$, a value greater than the expected duration of any of the three-state HMMs considered. We cannot calculate $\boldsymbol{\alpha}$ for all possible sequences, but using sequences less than $N$ time steps provides a good approximation.

For measuring the similarity between two states, $h_a$ and $h'_b$, we use the inverse of their divergence. In [12], the KL-Divergence was used, but we have chosen to use the Cauchy-Schwarz divergence (CSD) because it has a closed form solution and has been shown to perform comparably to the KL-Divergence [13]. CSD is defined as

$$\text{CSD}(\mathbf{p}, \mathbf{q}) = -\log \frac{\sum_i p_i q_i}{\sqrt{\sum_i p_i^2 \sum_i q_i^2}}. \qquad (2)$$

Once the new acoustic units have been created, the lexicon needs to be defined in terms of them. Each acoustic unit is a cluster of context-dependent graphemes. A simple method of generating pronunciations using the new acoustic units is to map the original graphemes to the new acoustic units based on their surrounding context. This will result in a lexicon where each entry contains the same number of units as the original grapheme-based lexicon. New acoustic models are then trained based on the new lexicon. In the next section, we present a method for improving the pronunciations based on the new acoustic units.

| Source | Target | $p(t\|s)$ | Change in LLH |
|---|---|---|---|
| a c k | a k | 0.19 | -13.05 |
| c h s | c x s | 0.13 | 48.25 |
| c e s s a | s e s e | 0.36 | 63.28 |
| f o r d | f r d | 0.17 | -81.47 |
| a u g h t | o t | 0.25 | 87.39 |

**Table 1**. Example phrase table from Moses [16]. Each row shows a translation from the first column to the second column. $p(t|s)$ is the probability of translating the source into the target. The final value is the average change in log-likelihood as determined by our rule scoring procedure. Note that the phrase table produced by Moses contains additional values.

## 3. PRONUNCIATION TRANSFORMATION

We assume that we have an initial dictionary which contains systematic errors; this initial dictionary could be a hand-crafted dictionary, a grapheme-based lexicon, or based on the discovered acoustic units described in the previous section. To simplify the discussion, we describe the approach in terms of graphemes, but the discovered acoustic units from Section 2 can also be used. Our goal is to transform the pronunciations such that acoustic models trained using the new lexicon will be improved and result in reduced WER. Our approach is similar to statistical machine translation (SMT)-based approaches to grapheme-to-phoneme (G2P) conversion [14, 15]. Since our transformation approach uses the same label set for input and output, we will refer to the system as a grapheme-to-grapheme (G2G) system. An initial grapheme decoder generates pronunciations hypotheses for every word in the training set. An SMT-based G2G system is trained based on the hypothesized pronunciations. The rules used in the G2G system are pruned and scored. Finally, the improved G2G system is used to transform the original pronunciation dictionary.

Using a previously defined lexicon, a set of context-independent models—one model per grapheme—is trained; due to the smaller number of models compared to a context-dependent system, we use 128 mixtures per GMM. While a context-dependent model would likely produce more accurate grapheme recognition in terms of the original pronunciation lexicon, our purpose is to transform the original lexicon. Also, the sequence of possible graphemes would be artificially restricted by the contexts seen during training.

Each pronunciation hypothesis becomes a training example for the SMT-based G2G system. The canonical pronunciation is used as the source language and the hypothesis is used as the target language. For our experiments, we use the Moses [16] toolkit. Moses builds a phrase translation table based on the aligned training data. For our purposes each phrase would represent the translation of a sequence of graphemes into an alternate sequence of graphemes. Examples are shown in Table 1. The third column represents the probability of translating the source language into the target, one of several probabilities automatically calculated by Moses.

It is important to note that the training data we provide to Moses is very noisy; many of the rules would result in significantly worse performance for the recognition system. Typically, the next step is to tune Moses on a held-out dataset using discriminative training. Unfortunately, we do not have access to such data as there is no gold standard for the transformation we are computing. Instead we propose an alternative method for scoring each individual rule based on its effect on the likelihood of the training data. The intuition is that we can consider the original lexicon as being generated by a phrase table with only identity rules. We only want to introduce new rules that will improve over this original lexicon.

Each rule is evaluated individually. A new pronunciation dictionary is generated that differs from the baseline dictionary only by the application of a single phrase-based rule. Using the new dictionary, we force align the training data—using the context-independent models previously mentioned—and measure the average effect on the likelihood of each sentence. The average change in likelihood becomes the score for the rule. Note that a phrase table could contain over one million rules and scoring each rule individually would be prohibitively expensive. We first prune the phrase table by only scoring rules that are both common—the source phrase appears at least ten times in the training corpus—and contain between three and five graphemes on the source side; phrases with less than three graphemes likely do not contain enough context information to be useful and phrases with more than five graphemes may not generalize to unseen words. Returning to Table 1, the final column shows examples of scores generated by the just described procedure. In this sample, as in the remainder of the phrase table, there is no correlation between the probabilities generated by Moses and the scores based on the change in likelihood. This is not a fault of Moses, but an indication of the level of noise seen in the training data.

Once the rules have been scored, we only keep rules which surpass a certain threshold—chosen to maximize performance on the development set. This subset of rules becomes the final phrase translation table. Using Moses, the original pronunciation dictionary is transformed by the rules in the phrase table. New models are trained using the transformed pronunciation dictionary and are used for the final recognition system.

## 4. EXPERIMENTAL SETUP

We use the HMM toolkit (HTK) [17] for our recognition system. The acoustic model consists of cross-word triphones; each triphone has three states, modeled by a mixture of 16 Guassians per state. Transition probabilities are tied across all models with the same center symbol. Individual states are

| Unit Type | # Units | Direct | Transformed |
|---|---|---|---|
| Grapheme | 26 | 15.8 | 14.5 |
| Discovered | 39 | 15.0 | 13.9 |
| Discovered | 50 | 15.2 | 13.9 |
| Discovered | 60 | **14.4** | **13.8** |

**Table 2**. Results for both grapheme-based acoustic units and automatically discovered acoustic units (Section 2) in terms of WER (%). Direct are the original pronunciations while Transformed refers to our proposed pronunciation transformation approach (Section 3).

clustered across models, resulting in approximately 2000 tied states. For a standard phone-based system, state clustering is based on questions relating to phonetic classes. This information does not exist for the grapheme and automatically discovered models, so we use singleton questions (one question per grapheme) as is used in other work [3]. Decoding is performed with a bigram language model.

All evaluations are performed on the WSJ0 corpus, an English language 5000 word closed vocabulary task. The training set consists of 7,138 utterances from 83 speakers for a total of 14 hours of speech. The test set consists of 330 utterances from 8 speakers not seen during training. In this work, English was chosen because it allows for a comparison against using a hand-crafted dictionary and it is a difficult language for grapheme-based speech recognition [3].

The previously described acoustic unit discovery process is applied to the grapheme-based HMM models. Pronunciation hypotheses are generated from a separate set of context-independent models with 128 Gaussians per state. These hypotheses are separated into word pronunciations by using the word boundaries obtained from force aligning the training data with the baseline grapheme-based system. In total, about 100k pronunciation hypotheses are generated. The Moses system produces approximately 500k phrase-based rules, which are pruned down—as described in the previous section—to 25k rules.

## 5. RESULTS

Recognition results are presented in Table 2 in terms of WER. While the performance of the baseline grapheme-based system is significantly worse than a comparable phone-based system (8.0% WER), it is similar to previously published results on this dataset [3]. The first results column displays WER when using pronunciation dictionaries without the pronunciation transformation. The second column shows the improvement from using the pronunciation transformation (see Section 3).

The pronunciation transformation alone provides a reduction of 8% relative WER over the baseline grapheme-based system—row 1 in Table 2. Note that if the hypothesized pronunciations were used to directly train a standard G2P system [7], it would significantly decrease performance (18.7% WER). Results using the automatically discovered units (with $k$ = 39, 50, and 60) are also shown in Table 2. Note that the value of 39 was chosen to match the number of units used in the phone-based dictionary. Using the new acoustic units does provide some improvement—*Direct* column in Table 2—but the best result is obtained when combining the discovered acoustic units with the pronunciation transformation for a relative word error reduction of 13%. We should also emphasize that all systems use the same number of tied states, so the improved performance of the discovered acoustic units cannot be attributed to an increase in the number of parameters.

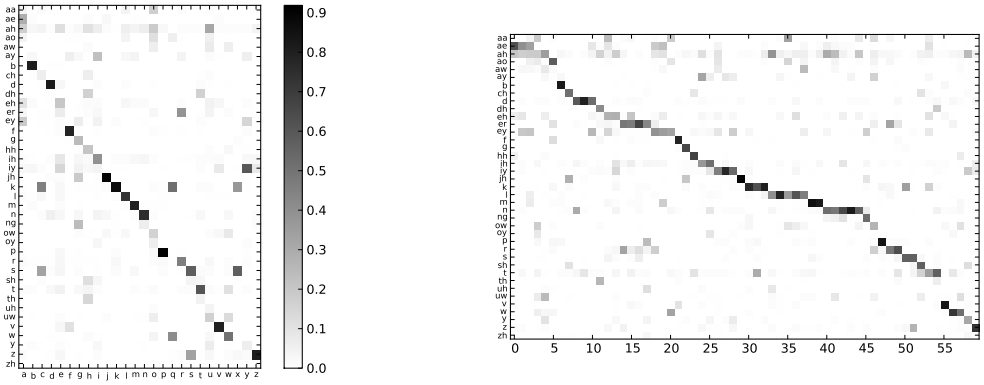
## 6. DISCUSSION

In this section we further analyze the performance of the acoustic units and pronunciation lexicons. We show how the learned acoustic units compare to phonetic units and how the pronunciation transformation improves the correlation. Since our metric for scoring the phrase-based rules uses context-independent models, we also analyze the relative performances of using context-independent models. Finally, computational considerations of our approach are addressed.

### 6.1. Comparison to Phonetic Units

Comparisons between phonetic units and the graphemic and discovered units used in this work are shown in Figure 1. Each subfigure shows the phones on the y-axis and either graphemes or discovered units on the x-axis. Since the plot is normalized by row, each point shows the percentage of frames a particular model corresponds to each phone. Graphemes appear to be only weakly correlated with phonemes. In particular, no strong relationships exist between any grapheme and any vowel. As expected, certain graphemes correspond to several phones (e.g., ‘q’ with /k/ and /w/, and ‘c’ with /ch/, /k/, and /s/). The automatically discovered units appear to have a stronger correlation—in large part due to the increased number of units. Certain phones (e.g. /ch/ and /sh/) that did not have models associated with them in the grapheme-based system have discovered acoustic units associated with them. Also, several models emerge to represent vowels in particular contexts; however, some vowels never see any models strongly associated with them (e.g. /ay/, /uh/). Note that while the pronunciation transformation improved overall performance, it did not visibly affect the plots in Figure 1.

Based on the relationship between orthography and phonetics in English, some phones would be nearly impossible to separate with our approach. For instance /dh/ and /th/ share a single model. Since both phones are typically represented by the same series of graphemes, the spectral clustering method has no means of separating them. Vowels can also be difficult because their pronunciation is not always determined by



(a) Grapheme

(b) Discovered 60

**Fig. 1**. Correlation between phone models and graphemes, and between phone models and discovered models. In each case the phones are listed on the y-axis. Note that each column is normalized by its sum. (a) Grapheme. (b) Discovered 60. Please see Section 6.1 for discussion.

| Unit Type | # Units | Direct | Transformed |
|---|---|---|---|
| Grapheme | 26 | 39.9 | 31.4 |
| Discovered | 39 | 34.3 | 26.7 |
| Discovered | 50 | 31.6 | 25.8 |
| Discovered | 60 | **29.3** | **23.3** |

**Table 3**. Results for context-independent models in terms of WER (%). Note that in this case, the transformed pronunciations still use the originally trained models; only the pronunciation dictionaries have changed.

their immediately surrounding graphemic context. Consider the vowels /ae/ and /ey/ in the words *fat* and *fate* respectively; both phones are represented by the same context-dependent model. In Figure 1, any model associated with /ae/ is also associated with /ey/.

### 6.2. Context Independent Performance

Table 3 shows results using the same pronunciation dictionaries and acoustic units as Table 2, but recognition is performed with context-independent models. The gap between the baseline grapheme-based system and the learned acoustic units with transformed lexicons is much greater. Our approach appears to create acoustic units that are more stable across contexts. These large improvements are decreased when context-dependent models are used. At least part of the gain seen through using the discovered units in the context-independent results can be attributed to the increased number of total models, but that does not explain the further improvements produced by the pronunciation transformation.

As has been noted previously [18], context-dependent models do a good job of capturing phonetic variation. Since our approach to acoustic unit discovery started from context-dependent models, the new units may have been clustered around variations that were already well captured by the context-dependent graphemes. The improvements provided by the pronunciation transformation may have largely been due to deleting acoustically unrealized units (e.g. the silent 'e' in ice) or inserting additional units, situations that are not well handled by context-dependent models.

### 6.3. Computational Considerations

The computation required for the acoustic unit discovery consists of two main components, computation of the similarity matrix and performing spectral clustering. Due to our efficient formulation for computing the similarity between two HMMs, it takes only two hours to compute the pairwise similarity between 4000 HMMs on a single 2.0 GHz processor. Performing spectral clustering on the similarity matrix takes an additional 30 minutes on the same processor.

Computing the pronunciation hypotheses requires a single decoding pass through the training data and can be performed in parallel. With approximately 100k total hypotheses, the time required by Moses to compute the phrase table is trivial. The majority of the computation required by our approach is in scoring the phrase-based rules. We increase the speed by only force-aligning sentences where a pronunciation for a word has changed and limiting the total number of sentences. Scoring each rule takes at most 30 seconds, and the process can be sped up through parallelization.

## 7. CONCLUSIONS

We have presented a method for discovering acoustic units and learning a corresponding pronunciation lexicon from an

initial grapheme-based lexicon. Both the acoustic unit discovery and the pronunciation transformation individually produce a significant improvement over a grapheme-based baseline; combined, they further reduce the WER. As opposed to prior work, our clustering approach works on full HMM models instead of individual HMM states. Our pronunciation transformation method demonstrates a method for introducing acoustic-based scores that does not exist in other approaches.

While our results are presented on an English dataset, we believe our framework could be used with low-resource languages without an existing lexicon, such as in the Babel project. In the future, we will work to improve both the acoustic unit discovery and pronunciation transformation components of our framework. In addition, we will apply our methods to low-resource languages.

### 8. ACKNOWLEDGEMENTS

This research was partially supported by OSEO, the French State agency for innovation, under the Quaero program and by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Defense US Army Research Laboratory contract number W911NF-12-C-0013. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoD/ARL, or the U.S. Government.

## References

[1] A. Jansen and K. Church, "Towards unsupervised training of speaker independent acoustic models," in *Proceedings of Interspeech*, 2011, pp. 1693–1696.

[2] C. Lee and J. Glass, "A nonparametric bayesian approach to acoustic model discovery," in *Proceedings of the Association for Computational Linguistics*, 2012, pp. 40–49.

[3] M. Killer, "Grapheme-based speech recognition," M.S. thesis, Carnegie Mellon University, 2003.

[4] B. Varadarajan, S. Khudanpur, and E. Dupoux, "Unsupervised learning of acoustic sub-word units," in *Proceedings of the Association for Computational Linguistics*, 2008, pp. 165–168.

[5] M. Bacchiani and M. Ostendorf, "Joint lexicon, acoustic unit inventory and model design," *Speech Communication*, vol. 29, no. 2, pp. 99–114, 1999.

[6] R. Kaplan and M. Kay, "Regular models of phonological rule systems," *Computational Linguistics*, vol. 20, no. 3, pp. 331–378, 1994.

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

[8] N. Goel, S. Thomas, M. Agarwal, P. Akyazi, L. Burget, K. Feng, A. Ghoshal, O. Glembek, M. Karafiat, D. Povey, A. Rastrow, R. C. Rose, and P. Schwarz, "Approaches to automatic lexicon learning with limited training examples," in *Proceedings of IEEE ICASSP*, 2010, pp. 5094–5097.

[9] S. Kanthak and H. Ney, "Context-dependent acoustic modeling using graphemes for large-vocabulary speech recognition," in *Proceedings of IEEE ICASSP*, 2002, pp. 845–848.

[10] U. Von Luxburg, "A tutorial on spectral clustering," *Statistics and Computing*, vol. 17, no. 4, pp. 395–416, 2007.

[11] A. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," *Advances in Neural Information Processing*, vol. 14, pp. 849–856, 2002.

[12] S. M. E. Sahraeian and B. J. Yoon, "A novel low-complexity hmm similarity measure," *IEEE Signal Processing Letters*, vol. 18, no. 2, pp. 87–90, 2011.

[13] K. Kampa, E. Hasanbelieu, and J. C. Principe, "Closed-form cauchy-schwarz pdf divergence for mixture of gaussians," in *Proceedings of IEEE IJCNN*, 2011, pp. 2578–2585.

[14] Panagiota Karanasou and Lori Lamel, "Pronunciation variant generation using SMT-inspired approaches," in *Proceedings of IEEE ICASSP*, 2011, pp. 4908–4911.

[15] A. Laurent, P. Deléglise, and S. Meignier, "Grapheme to phoneme conversion using an SMT system," in *Proceedings of Interspeech*, 2009, pp. 708–711.

[16] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst, "Moses: Open source toolkit for statistical machine translation," in *Proceedings of the Association for Computational Linguistics*, 2007, pp. 177–180.

[17] S. Young, G. Evermann, T. Hain, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland, *The HTK Book*, Cambridge University Publishing Department, 2002.

[18] D. Jurafsky, W. Ward, Z. Jianping, K. Herold, Y. Xiuyang, and Z. Sen, "What kind of pronunciation variation is hard for triphones to model?," in *Proceedings of IEEE ICASSP*, 2001, pp. 577–580.