INTERSPEECH 2016
September 8–12, 2016, San Francisco, USA

Automatic Classification of Lexical Stress in English and Arabic Languages using Deep Learning

Mostafa Shahin [1], *Julien Epps*[2], *Beena Ahmed* [1]

[1]Dept.of Electrical and Computer Engineering, Texas A&M University, Doha, Qatar
[2]The School of Electrical Engineering and Telecommunications, University of New South Wales
{Mostafa.shahin, beena.ahmed}@qatar.tamu.edu, j.epps@unsw.edu.au

Abstract

Prosodic features are important for the intelligibility and proficiency of stress-timed languages such as English and Arabic. Producing the appropriate lexical stress is challenging for second language (L2) learners, in particular, those whose first language (L1) is a syllable-timed language such as Spanish, French, etc. In this paper we introduce a method for automatic classification of lexical stress to be integrated into computer-aided pronunciation learning (CAPL) tools for L2 learning. We trained two different deep learning architectures, the deep feedforward neural network (DNN) and the deep convolutional neural network (CNN) using a set of temporal and spectral features related to the intensity, duration, pitch and energies in different frequency bands. The system was applied on both English (kids and adult) and Arabic (adult) speech corpora collected from native speakers. Our method results in error rates of 9%, 7% and 18% when tested on the English children corpus, English adult corpus and Arabic adult corpus respectively.

Index Terms: lexical stress detection, deep neural network, convolutional neural network, Arabic lexical stress

1. Introduction

In stress-timed languages such as English and Arabic, at least one syllable should be stressed relative to the other syllables in any multi-syllabic word. This relative stress in a word is known as *lexical stress*. Lexical stress plays an important role in the perception and processing of speech by native speakers. In addition, shifting the placement of the stress in the word can change the word's meaning. The so called "stress-minimal word pairs" are defined as a pair of words which are phonetically identical and different only in the position of the stressed syllable such as "PERfect" (noun) and perFECT" (verb). These types of words exist also in Arabic but are less frequent than English, e.g. "WAsafa, وَصَفَ" where stress is placed on the first syllable and means "he described" and the word "wasaFA, وَصَفَى" which means "and it cleared up" [1].

The position of the stressed syllable in English is unpredictable and need to be defined and learned for each individual word, therefore English pronunciation dictionaries include the position of the stress as a part of the word's pronunciation. Unlike English, lexical stress in Standard Arabic (SA) follows regular rules, which depend on the syllabic structural of the multi-syllabic words [2]. SA syllables can be classified as either "open syllables", which end with a vowel (CV, CVV) and "closed syllables", which end with a consonant (CVC, CVVC, CVCC). The stress rules in SA can be summarized as follows: 1) the last syllable in the word is stressed if it is super-heavy, i.e. has more than 3 units of time (CVVC, CVCC), 2) the second syllable from the end (penult) is stressed if it is heavy, contains 3 units of time (CVC, CVV), and 3) otherwise the stress is placed on the third syllable from the end (antepenult) and never falls further back [3]. Despite the existence of these rules, there is inconsistency in the lexical stress among the speakers from different regions in the Arab world [4].

Native syllable-timed language speakers trying to acquire a stress-timed language find it difficult to change stress levels within a word. Moreover, due to the variation in lexical stress rules among different stress-timed languages, speakers may apply their native language rules on the acquired language and lead to unintelligible pronunciation. In this paper, we thus propose a system to accurately detect lexical stress for computer-aided pronunciation learning (CAPL) tools that can be used by L2 learners to identify where to place stress in their productions.

CAPL tools facilitate L2 acquisition by allowing learners to learn and practice target languages at their convenience. Most state-of-the-art CAPL tools have an automatic assessment feature that aims to give direct feedback to learners about the quality of their pronunciation. The automatic assessment of lexical stress is an important component of measuring the quality of the speaker's pronunciation, given its impact on intelligibility.

There has been significant work on the classification of lexical stress especially for native and non-native English speakers. Most of these methods are based on temporal features such as intensity, duration, pitch, etc [5, 6], with the use of spectral features restricted. In [7] the author obtained best performance when both temporal and spectral features were combined. Different types of classification methods have been used to detect lexical stress e.g. support Vector Machines (SVM) [5, 8], Hidden Markov Model (HMM) [9], Maximum Entropy (MaxEnt) [10] but deep learning algorithms have seen limited application [6]. In [11] Ferrer et. al., trained 3 different classifiers, Gaussian mixture models (GMMs), a decision tree and a neural network to estimate the posterior probabilities of syllabic stress level in a children corpora consisting of both native and non-native English speakers and obtained accuracy of 88.5% and 79.8% on native and non-native datasets respectively. Despite all the work on lexical stress detection in English, there has been little interest on working in other low resource languages such as Arabic. The few systems developed for Arabic lexical stress detection include that by Chentir et al, where the authors used discriminate analysis to detect the stressed syllable in SA

http://dx.doi.org/10.21437/Interspeech.2016-644

words of structure (CVCVCV) produced by only four native speakers [12].

In this paper, we compare the performance of two deep learning classifiers, the deep feedforward neural network (DNN) and the deep convolutional neural network (CNN), used to classify syllabic stress level. We performed this comparison to determine whether recent improvements using the CNN over the DNN on other classification problems can be replicated here [13]. Our system was tested against two speech corpora collected from native English speakers, one children's corpus and the standard TIMIT corpus, and one corpus collected from Arabic speakers.

The rest of the paper is organized as follows. The methods used for feature extraction, the classifier architecture and the speech corpus used are provided in Section 2. The experiments conducted and results obtained are detailed in Section 3 and finally the conclusions presented in Section 4.

2. Method

2.1. System description

Figure 1 presents a flowchart of our system. The first step is to determine the time boundaries of each pronounced phoneme in order to extract acoustic features from each syllable. As the pronounced phoneme sequence is given in all used speech corpora, a straightforward HMM-based force alignment is performed. The acoustic model was trained from the same corpus to reduce the segmentation error. A set of temporal and spectral features was then extracted from the speech signal for each syllable. Since the stressed syllable is defined as the most prominent syllable in the word, considering the neighbor syllables is important. Therefore, the feature vector extracted from the target syllable was then combined with the feature vectors of the first preceding and succeeding syllables.

The post processing block has two main functions: 1) to normalize all features' values to 1 and -1 and 2) fix the size of the feature vector by a frame selection/padding process as will be explained in the next section. The resulting features are then fed into two different classifiers, a DNN and a CNN, to classify the stress level of each syllable as primary-stress (PS), secondary-stress (SS) or no-stress (NS) or combining PS and SS in one class (S) and then binarily classifying each syllable's stress level as stress (S) or no-stress (NS).

2.2. Feature Extraction

Lexical stress is identified by the variation in the pitch, energy and duration produced between different syllables in a multi-syllabic word [14]. The stressed syllable is characterized by increased energy and pitch as well as a longer duration compared to the other syllables within the same word. Therefore we extracted seven features $(f_1 - f_7)$ related to these characteristics as listed in Table 1.

Table 1. *The extracted acoustic features*

Feature	Description
f_1	Peak-to-peak amplitude over syllable nucleus
f_2	Mean energy over syllable nucleus
f_3	Maximum energy over syllable nucleus
f_4	Nucleus duration
f_5	Syllable duration
f_6	Maximum pitch over syllable nucleus
f_7	Mean pitch over syllable nucleus
f_8	27 Mel-scale energy bands over syllable nucleus

These seven features are commonly used in the detection of the stressed syllable in a word [8, 15-17]. As the speech signal energy is distributed over different frequency bands, we also computed the energy in the Mel-scale frequency bands in each frame of the syllable nucleus. The speech signal was divided into 10 msec non-overlapped frames and the energy, pitch and the frequency bands energies calculated for each frame. Each syllable thus had 7 scalar values representing energy, pitch and duration $(f_1 - f_7)$ and $27 * n$ Mel-coefficients where n is the number of frames in each syllable's vowel. All features extracted using "praat" software [18]. To handle variable vowel lengths, we limited the number of input frames provided to the classifier to N frames for each syllable. If $n > N$, the middle N frames from the vowels were selected and if $N < n$, (N – $(N - n)/2$ zero frames padded on each side (i.e zero padding). The resulting feature vector size was equal to $(27 \times N + 7) \times 3$, given we used a window of 3 consecutive syllables.

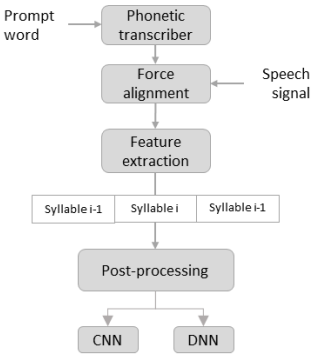

Figure 1. *System flowchart*

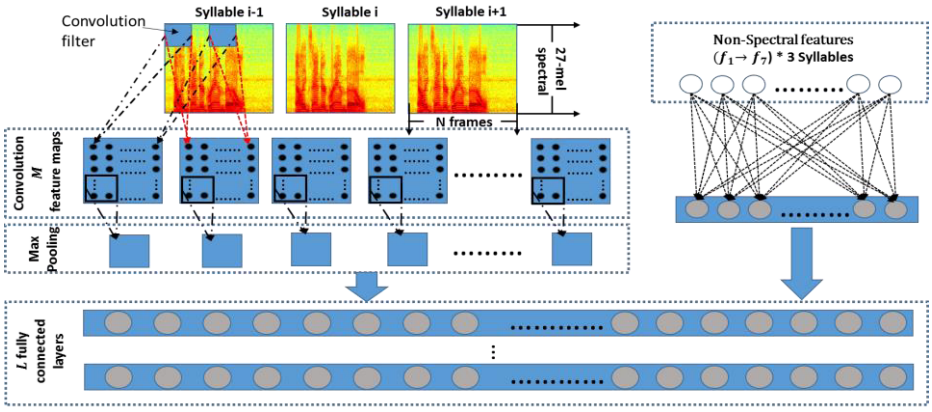

Figure 2. *The structure of the CNN. Three input feature maps represent the mel-spectral energies over N frames of the syllable's vowel. One convolutional layer consists of M feature maps and pooling layer with size p×p. The non-spectral inputs pass through one hidden layer and the output of this layer concatenated with the output from the pooling layer and fed to L fully connected hidden layers.*

2.3. Classifiers

In the past few years the use of deep learning has increased rapidly in a variety of classification problems. Due to increased availability of GPU units, it is now achievable to train very deep neural network architectures. In [19] the DNN has been successfully used in a speech recognition problem and showed to outperform the traditional HMM model when applied on the TIMIT database. Following the success of the DNN, the CNN has also been used to address the same problem and reported promising results [13]. However, the use of deep learning algorithms in the context of lexical stress detection is still very limited. In [16] a deep belief network was trained to classify the syllable stress in L2 English speech datasets and achieved an accuracy of 80%. A deep neural network was used in [11] to estimate the posterior probabilities of the stress classes. In our previous work [20] we used the DNN in the classification of bisyllabic lexical stress patterns in disordered speech and obtained an overall accuracy of 74%.

In this work we used two different deep learning architectures, the feedforward fully connected DNN and the deep CNN with one convolutional layer and multiple fully connected hidden layers to classify the syllable stress level. The DNN was trained directly from the extracted features. The input layer receives the features of the 3 syllables window as a one wide feature vector. The target output was set to be the class of the middle syllable stress level. The number of hidden layers and units in each layer was tuned empirically.

For the CNN, the input feature vector was organized to fit the structure of the CNN [21]. As shown in figure 2 the input layer consisted of three 2-D feature maps representing the 27 mel-spectral energies over the N frames of the vowels in the 3 consecutive syllables. A convolutional layer with a tunable number of feature maps followed this input layer. Each unit in the convolutional feature maps received inputs from a local-region of each of the input feature maps called the receptive field, determined by a 2-D filter. This filter scans each input feature map and shares the same weight matrix for each feature map. The dimensions of the filter are also tuning parameters. This allows the features to be modeled regardless of their actual position in the input feature space. The convolutional layer was followed by a max-pooling layer that works as a down-sampler. Each feature map in the convolutional layer was divided into non-overlapping sub-regions and output the maximum value from each region. The max-pooling layer reduces the dimensionality of the previous layer by $1/p$ where $p \times p$ is the dimension of the sub-region, another tuning parameter.

The other features $(f_1 - f_7)$ are non-spectral and thus cannot be treated similarly. Therefore, all the seven features from the 3 consecutive syllables are inputted to a single hidden layer. The output of this layer and the output of the max-pooling layer path are fed to a series of L fully connected hidden layers. The final layer of both the DNN and the CNN is a softmax layer with number of output equal to the number of target classes. Both classifiers are trained using the mini-batch stochastic gradient decent method (MSGD) with adaptive learning rate. The learning rate starts with an initial value (typically 0.1) and after each epoch the loss in the error of the validation data set is computed. If the loss is greater than zero (i.e. the error increases) the training continues with the same learning rate. If the error continues increasing for 10 consecutive epochs, the learning rate is halved and the parameter of the classifier returned to the one that achieved minimum error. Training is terminated when the learning rate reaches its minimum value (typically 0.0001) or after 200 epochs, whichever is earlier. The performance of the classifiers is then computed using a separate testing set. The "Theano" library was used for the implementation of the classifiers [22].

2.4. Speech corpus

The system was applied on three different speech corpora to determine the results were consistent across different domains. The first one is the OGI children speech corpus [23]. This corpus was collected from 1100 native English children from grad 0 to 10. Each child pronounced around 200 prompted isolated words and 100 prompted sentences. This corpus is very similar to the one used in [11]. We will refer to this corpus as the "kids" corpus. The speakers in this corpus were divided into 80% for training, 10% for validation and 10% for testing. All the grades appeared in the training, validation and testing sets with the same ratio.

In addition to the "kids" corpus we also used the standard TIMIT dataset which was collected from native English adult male and female speakers distributed over 8 different dialect regions. We used the training and testing sets as suggested in the corpus document [24].

The last corpus used was the Arabic speech corpus (AR). This corpus consists of 6 hours of recordings obtained from the Arabic language learning program "alkittaab text book" [25]. All the data was recorded from native Arabic speakers (male and female) and in Modern Standard Arabic (MSA).

The stress levels in the two English speech corpora (kids, TIMIT) are assigned automatically using the CMU pronunciation dictionary as both are collected from native speakers. The stress levels for the Arabic data were marked manually by an Arabic linguist as there is no available lexical stress dictionary for Arabic and the lexical stress rules followed by Arabic speakers depends on their native dialect.

3. Experiments and results

In all the experiments the number of samples in each class was kept equal in the training, validation and testing sets to guarantee the balance of the system.

In the first experiment we trained both DNN and CNN classifiers to discriminate between the three stress levels (PS/SS/NS) using the "kids" corpus. As this corpus is the biggest corpus we have, we used it to show the effect of the tuning parameters in different classifiers. The number of frames N was first tuned and the best accuracy obtained at $N = 30$ frames. The number of frames was then fixed at this value and the other two parameters changed. Figure 3 shows the error rate of the DNN classifier as a function of the number of hidden layers and the number of hidden units per layer at $N = 30$ frames. As shown in the figure, the error decreased from 11.8% to 10.4% when moving from a shallow network with one layer to 2 layers with 500 units each. The best error rate of 9.9% was obtained at 3 layers with 300 units per layer with a fewer number of parameters compared to the 2 layers with 500 units.

We then repeated the same experiment with the CNN. There are 6 tuning parameters for the CNN, the number of input frames (N), the number of feature maps in the convolution layer (M), the convolution filter size, the pooling

size (p), the number of fully connected hidden layers (L) and the number of hidden units in each layer. Figure 4 shows the effect of the number of feature maps (M) and the pooling size on the classification error rate. The input mel-coefficients extracted from 35 vowel's frames ($N = 35$), the filter size is fixed on 3×3 and the number of fully connected hidden layers and the number of hidden units per layers are fixed to 3 and 1000 respectively.

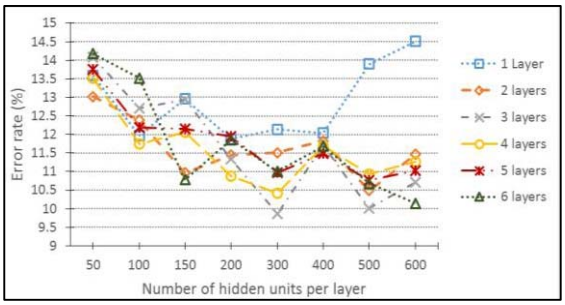

Figure 3: *The effect of different number of hidden layers and hidden units per layer on the error rate of the "kids" data using DNN. The number of frames (N) fixed at 30 frames.*

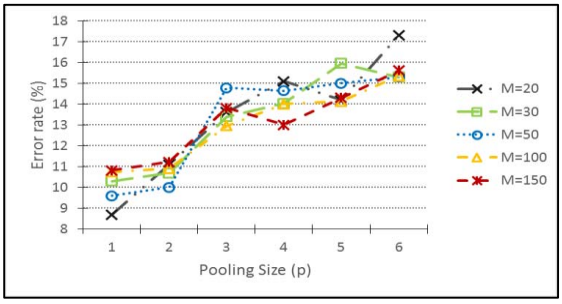

Figure 4. *The CNN error rate of the "kids" data as a function of the pooling size (p) and number of feature maps (M).*

The results show that the minimum error rate for all values of feature maps was obtained at $p = 1$, i.e. with no pooling. Increasing the pooling size to 2 slightly degrades the performance with a 75% reduction in the number of parameters. It is obvious that increasing the pooling size to 3 dramatically increase the error rate. We tried to perform the pooling over frequency bands only and over the time frames by using pooling windows of $(p \times 1)$ and $(1 \times p)$ respectively but we obtained a similar trend. Although the pooling step is one of the important feature of the CNN where it smartly reduces the complexity of the system and overcomes the overfitting problem, it does not help in this problem. This can be explained by the nature of the kids corpus, which consists of a limited number of words/short sentences repeated by a large number of speakers. The same words appear in training, validation and testing sets and therefore the same lexical stress patterns and syllable structure. This leads to overfitting around these patterns and thus the actual location of the feature becomes important and is lost in the down sampling process. Even though this overfitting may affect the generalization of the system, most CAPL tools contain a limited number of prompt words/sentences and customizing the model to perform well on these words/sentences is desirable. We then tested the system against the TIMIT and AR data. As the TIMIT data contains a limited number of secondary stressed syllables, we trained the classifiers to discriminate between stressed (S) and unstressed (NS) syllables. The manual annotation of the Arabic data marked each syllable as either stressed or unstressed resulting intwo classes in the AR data.

Table 2. *Summary of the classification error rates*

Dataset	Classifier	Classes	Error rate (%)			
			PS	SS	NS	Overall
Kids	DNN (3 layers, 300 units)	PS/SS/NS	12.6	5.5	11.7	9.85
Kids	CNN (no pooling)	PS/SS/NS	11.9	6	11.2	**8.7**
Kids	DNN (5 layers, 300 units)	S/NS	8.1		6.3	7.2
Kids	CNN (no pooling)	S/NS	7.2		6	**6.52**
TIMIT	DNN (2 layers, 200 units)	S/NS	5.4		10.1	7.7
TIMIT	CNN (p =2)	S/NS	5		9.8	**7.2**
AR	DNN (3 layers, 150 units)	S/NS	16.9		19.4	**17.9**
AR	CNN (p=4)	S/NS	17.5		18.6	18

Table 2 summarizes the results of all experiments. The CNN outperforms the DNN in both the "kids" and TIMIT data sets and gives roughly the same accuracy as the DNN in the Arabic data set.

Unlike the "kids" dataset, both TIMIT and AR benefit from the pooling step due to the sparsity and small amount of data relative to the "kids" data. The CNN best accuracy for the TIMIT data was obtained using a filter size of 3x3, with 30 feature maps and 2 fully connected hidden layers with 300 units each while the parameters of the CNN in AR data were a 3x3 filter size, 20 feature maps and 5 fully connected hidden layers with 200 units each.

This degradation in the performance of the AR may be due to inaccurate manual assessment as each word was assessed by only one linguistic. Obtaining marking from a second linguistic should help in measuring the inconsistency of the perceptual assessment of the lexical stress in Arabic. Moreover, a deeper analysis of the results is needed to understand the difficulties in the classification problem.

4. Conclusions

We have proposed an automatic lexical stress detection system for English and Arabic native speech. Our comparison of the DNN and CNN classifiers to discriminate between 3 syllabic stress levels (PS, SS, NS) has shown that the CNN outperforms the DNN in most of the experiments as in previous work [13]. To our knowledge this is the first intensive work on the automatic detection of lexical stress in Arabic, with other work focusing only on one syllabic structure [12]. The system achieved an overall accuracy of 91.3%, 92.8% and 82% when applied on native English children and adult speech and native Arabic adult speech respectively. In [11] an accuracy of 88.5% has been achieved on native English children corpus very similar to our corpus.

5. References

[1] S. Boudelaa and M. Meftah, "Cross-language effects of lexical stress in word recognition: the case of Arabic English bilinguals," in *Spoken Language, 1996. ICSLP 96. Proceedings., Fourth International Conference on*, 1996, pp. 121-124.

[2] S. H. Al-Ani, *Arabic phonology: An acoustical and physiological investigation* vol. 61: Walter de Gruyter, 1970.

[3] K. C. Ryding, *Arabic: A linguistic introduction*: Cambridge University Press, 2014.

[4] J. McCarthy and A. Prince, "Prosodic morphology and templatic morphology," in *Perspectives on Arabic linguistics II: papers from the second annual symposium on Arabic linguistics*, 1990, pp. 1-54.

[5] J.-Y. Chen and L. Wang, "Automatic lexical stress detection for Chinese learners' of English," in *Chinese Spoken Language Processing (ISCSLP), 2010 7th International Symposium on*, 2010, pp. 407-411.

[6] K. Li, X. Qian, S. Kang, and H. Meng, "Lexical stress detection for L2 English speech using deep belief networks," in *INTERSPEECH*, 2013, pp. 1811-1815.

[7] N. Chen and Q. He, "Using nonlinear features in automatic English lexical stress detection," in *Computational Intelligence and Security Workshops, 2007. CISW 2007. International Conference on*, 2007, pp. 328-332.

[8] J. Zhao, H. Yuan, J. Liu, and S. Xia, "Automatic lexical stress detection using acoustic features for computer assisted language learning," *Proc. APSIPA ASC,* pp. 247-251, 2011.

[9] C. Li, J. Liu, and S. Xia, "English sentence stress detection system based on HMM framework," *Applied Mathematics and Computation,* vol. 185, pp. 759-768, 2007.

[10] V. Rangarajan, S. Narayanan, and S. Bangalore, "Exploiting acoustic and syntactic features for prosody labeling in a maximum entropy framework," in *Proceedings of NAACL HLT*, 2007, pp. 1-8.

[11] L. Ferrer, H. Bratt, C. Richey, H. Franco, V. Abrash, and K. Precoda, "Classification of lexical stress using spectral and prosodic features for computer-assisted language learning systems," *Speech Communication,* vol. 69, pp. 31-45, 2015.

[12] A. Chentir, M. Guerti, and D. Hirst, "Discriminant Analysis for Classification of Stressed Syllables in Arabic," in *Proceedings of the World Congress on Engineering*, 2009, pp. 1-4.

[13] O. Abdel-Hamid, A.-r. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu, "Convolutional neural networks for speech recognition," *Audio, Speech, and Language Processing, IEEE/ACM Transactions on,* vol. 22, pp. 1533-1545, 2014.

[14] R. Kager, "Feet and metrical stress," *The Cambridge handbook of phonology,* pp. 195-227, 2007.

[15] Y.-J. Kim and M. C. Beutnagel, "Automatic assessment of American English lexical stress using machine learning algorithms," in *SLaTE*, 2011, pp. 93-96.

[16] K. Li, X. Qian, S. Kang, and H. Meng, "Lexical stress detection for L2 English speech using deep belief networks."

[17] J. Tepperman and S. Narayanan, "Automatic Syllable Stress Detection Using Prosodic Features for Pronunciation Evaluation of Language Learners," in *ICASSP*, 2005.

[18] P. Boersma, "Praat, a system for doing phonetics by computer," *Glot international,* vol. 5, pp. 341-345, 2002.

[19] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly*, et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *Signal Processing Magazine, IEEE,* vol. 29, pp. 82-97, 2012.

[20] M. Shahin, R. Gutierrez-Osuna, and B. Ahmed, " CLASSIFICATION OF BISYLLABIC LEXICAL STRESS PATTERNS IN DISORDERED SPEECH USING DEEP LEARNING," presented at the ICASSP, Shanghai, China, 2016.

[21] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE,* vol. 86, pp. 2278-2324, 1998.

[22] T. T. D. Team, R. Al-Rfou, G. Alain, A. Almahairi, C. Angermueller, D. Bahdanau*, et al.*, "Theano: A Python framework for fast computation of mathematical expressions," *arXiv preprint arXiv:1605.02688,* 2016.

[23] K. Shobaki, J.-P. Hosom, and R. Cole, "The OGI kids' speech corpus and recognizers," in *Proc. of ICSLP*, 2000, pp. 564-567.

[24] J. S. Garofolo, "Getting started with the DARPA TIMIT CD-ROM: An acoustic phonetic continuous speech database," *National Institute of Standards and Technology (NIST), Gaithersburgh, MD,* vol. 107, 1988.

[25] K. Brustad, M. Al-Batal, and A. Al-Tonsi. (2014). *Al Kitaab*. Available: https://www.alkitaabtextbook.com