Inteligencia Artificial 21(62), 134-144
doi: 10.4114/intartif.vol21iss62pp134-144

INTELIGENCIA ARTIFICIAL

http://journal.iberamia.org/

Artificial Neural Network (ANN) in a Small Dataset to determine Neutrality in the Pronunciation of English as a Foreign Language in Filipino Call Center Agents

Rey Benjamin M. Baquirin[1], Proceso L. Fernandez Jr.[2]
[1,2]Ateneo de Manila University, Quezon City, Philippines
[1]rey.baquirin@obf.ateneo.edu
[2]pfernandez@ateneo.edu

Abstract Artificial Neural Networks (ANNs) have continued to be efficient models in solving classification problems. In this paper, we explore the use of an ANN with a small dataset to accurately classify whether Filipino call center agents' pronunciations are neutral or not based on their employer's standards. Isolated utterances of the ten most commonly used words in the call center were recorded from eleven agents creating a dataset of 110 utterances. Two learning specialists were consulted to establish ground truths and Cohen's Kappa was computed as 0.82, validating the reliability of the dataset. The first thirteen Mel-Frequency Cepstral Coefficients (MFCCs) were then extracted from each word and an ANN was trained with Ten-fold Stratified Cross Validation. Experimental results on the model recorded a classification accuracy of 89.60% supported by an overall F-Score of 0.92.

Keywords: Automatic Speech Classification, Artificial Intelligence, Neural Networks, Mel-Frequency Cepstral Coefficients, Machine Learning

1 Introduction

As spoken language proficiency remains to be the most valuable skill call centers look for in their employees, the use of technologies that aid in its assessment and training have become a norm especially in the Philippine Business Process Outsourcing (BPO) industry [1]. Pearson's Versant and Berlitz's Spoken Language Test for example, are the most commonly used technologies that companies administer to screen prospective employees in language comprehension and speaking proficiency, including pronunciation [2].

Pronunciation is the act of producing sounds of speech in accordance to accepted standards of a language [3]. Published dictionaries indicate how words in a specific language should be pronounced to be properly understood by listeners. Nevertheless, these pronunciation guidelines are not rigidly followed as words that are deemed to have 'acceptable pronunciations' change in actual conversation and are affected by many factors including education, geography, social status, race, and culture [4]. This change leads to the acceptability of pronunciation to just be perceived as either 'neutral' or 'not neutral' depending on the circumstances surrounding the speakers. For most Filipino call centers, this pertains to their employees' ability to speak English in a way that is neutral enough for their clients to understand.

However, the technologies previously mentioned, Versant and Berlitz, are often costly and companies only ever utilize them once or twice per employee or applicant. Consequently, all other assessments conducted to measure spoken language proficiency rely on the individual knowledge and experience of recruiters and trainers leading to results with unavoidable personal bias. Call centers therefore suffer from having a subjectively varied standard for assessing spoken language proficiency, especially pronunciation.

ISSN: 1137-3601 (print), 1988-3064 (on-line)

This paper contributes to the knowledge space by proposing a model that can accurately classify whether call center agents' utterances are pronounced neutral or not. The goal is to train an Artificial Neural Network (ANN) that captures a uniform, objective standard for pronunciation neutrality assessment specific to a call center's standards albeit using a small dataset.

2 Literature Review

The decision to use an ANN for this study was drawn from the many publications that have used Machine Learning (ML) techniques in Automatic Speech Recognition (ASR). However, there is only a limited amount of literature regarding the use of ANNs to classify pronunciation neutrality as far as the researchers are aware of.

Studies like [5] used Deep Neural Networks (DNN) to detect mispronunciations of Mandarin and English words to enhance the performance of a Computer-Aided Language Learning (CALL) system. In addition, the authors of [6] used Hidden Markov Models (HMM) and DNNs to detect pronunciation errors of Japanese students learning Chinese to provide instructive feedback when using a Computer-Aided Pronunciation Training (CAPT) system. Both undertakings used ML for ASR but were more concerned on detecting pronunciation errors than generalizing whether an utterance is neutral or not. However, both also used Neural Network-based architectures as classifiers. Although HMM has been the most common model used in modern ASR systems, these studies exemplify that using Neural Networks as classifiers for speech or signal processing problems yield good results too.

[7] also focused on pronunciation but explored other areas. This research involved the use of crowd-sourcing techniques to generate pronunciations for named-entities. The study used the Google Voice Search production recognition engine, which runs on a DNN, to learn crowd-sourced business names and street names from a database of voice search queries in Google Maps. Rutherford and his team used a Grapheme-to-Phoneme dictionary mapping to facilitate pronunciation learning, which is a common technique used to evaluate the correctness of the model's predictions phonetically. For example, the string 'Iowa' can be mapped in the dictionary to be worded phonetically as [AY OW WUH] to successfully learn pronunciation. Like the previously mentioned studies, Neural Networks were also used as classifiers in this undertaking.

Adding to the fact that these studies used Neural Network-based classifiers, the use of Mel Frequency Cepstral Coefficients (MFCCs) as features for classifier training was also a noted commonality amongst them. MFCC extraction is one of the most used techniques in ASR research as it accurately approximates how humans generate speech sounds. Hence in this experiment, an ANN is trained entirely on MFCC features extracted from a small dataset of isolated speech utterances.

3 Methodology

Procedures, techniques, tools and algorithms used in the study are discussed in three sub-sections: Dataset Preparation, Feature Extraction, and Training and Validation.

3.1 Dataset Preparation

The dataset was collected from a Filipino call center catering to American clients. It contains 110 audio files, with each file representing one of ten words as follows: 'actually', 'basically', 'broadband', 'computer', 'Genie', 'internet', 'mobile', 'mobility', 'unfortunately', and 'wireless'. These words are the 10 most commonly used words in the call center where the data was collected. There are 11 utterances per word, with each utterance recorded from a different speaker.

Each utterance was recorded via Audacity and a condenser mic with 8000Hz sampling rate stored as 16-bit integers. All utterances were then saved as WAV files in a mono channel with the amplitude centred at 0dB. The trailing silences before and after the actual utterance were also removed in Audacity. The files have a typical duration of 0.3 – 1.5 seconds.

Ground truth labels of "Neutral" or "Not Neutral" were determined for each utterance with the help of two Learning Specialists as raters. The 110 utterances were labelled as 68 'Neutral' and 42 'Not Neutral' instances. To ensure the dataset's reliability and assess interrater agreement, Cohen's Kappa was computed as given by Equation 1.

$$\kappa \equiv \frac{p_o - p_e}{1 - p_e} = 1 - \frac{1 - p_o}{1 - p_e}, \tag{1}$$

where p_o is the observed proportionate agreement between the raters across all categories, and p_e is the probability that the raters will agree on a label [8].

Computations resulted to a kappa score of 0.82 for the dataset; showing strong agreement between the two raters and approximating 64-81% of the data as reliable to use as per table 1. The kappa score also reinforced the viability of the dataset to represent a baseline standard for pronunciation assessment specific to the company's requirements.

Table 1: Interpretation of Cohen's Kappa [9]

Value	Level of agreement	Sample of reliable data
0-0.20	None	0-0.4%
.21–.39	Minimal	4–15%
.40–.59	Weak	15–35%
.60–.79	Moderate	35–63%
.80–.90	Strong	64–81%
Above.90	Almost Perfect	82–100%

3.2 Feature Extraction

Feature vectors were extracted from each audio file in the form of Mel-Frequency Cepstral Coefficients (MFCCs) through Python. MFCC extraction is one of the most used techniques in ASR research as it accurately approximates how humans generate speech sounds [10]. MFCC extraction assumes that although a speech sound constantly changes over time, it can be represented by a series of power spectrums captured in very short time frames [11]. It was implemented in the study as per the process flow in figure 1.

Figure 1. The MFCC extraction process flow used in the study.

Each audio file was passed through a pre-emphasis filter to center the low and high frequency readings. It was then windowed using Hamming Windows with a window length of 25ms and a window step of 10ms, thereby framing the signal into short frames. Figure 2 shows an example of a windowed portion of an input audio file.

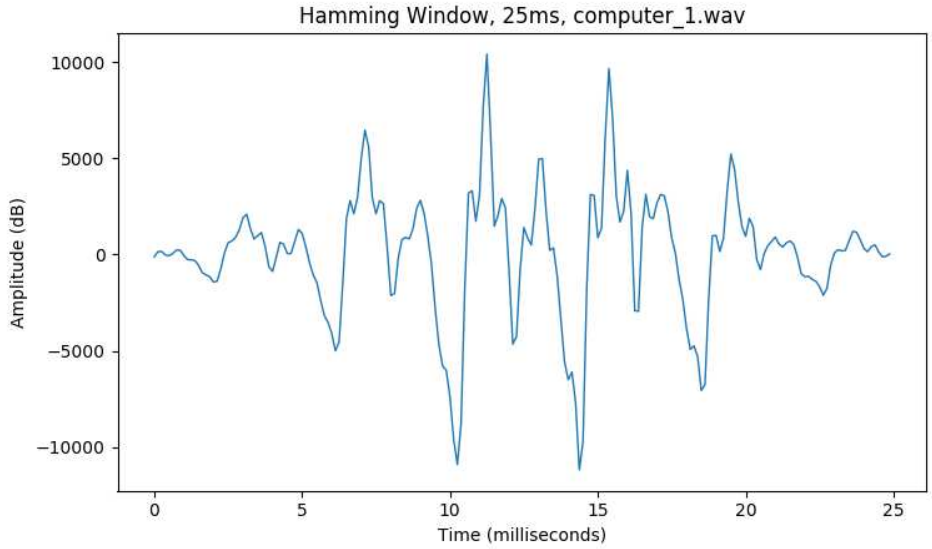

Figure 2. A 25ms hamming-windowed portion of the file 'computer1.wav'.

Windowing resulted to each frame of the input signal having 200 samples, derived from the original 8000Hz sample rate. The Fast Fourier Transform (FFT) algorithm was used in each frame to calculate spectral density, creating individual spectrums for each frame and approximating the periodogram of the power spectrum. Figure 3 shows the periodogram spectrum of figure 2 calculated through FFT.

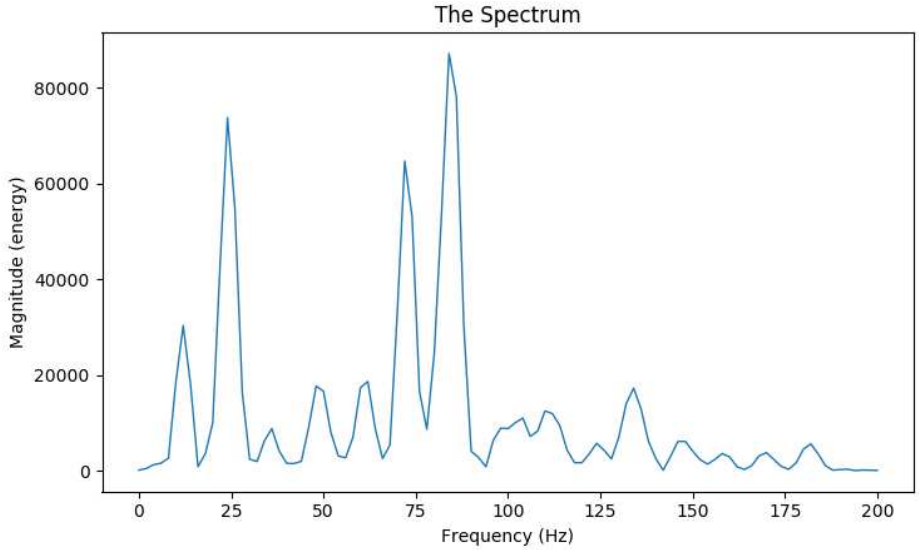

Figure 3. The spectrum generated after FFT.

The spectrum represents the identity of the input signal by detecting which frequencies are present in each frame [12]. To mimic human hearing and improve training results, a mel filterbank was used to discard information on higher frequency bands. This was done by spacing filters using the mel scale with more filters in lower to mid frequency bands as they are more relevant to human hearing and are where speech signals are commonly found [13]. Twenty-six (26) filters were used in the filterbank, separated across the spectrum via the computed mel scale as shown in figure 4.

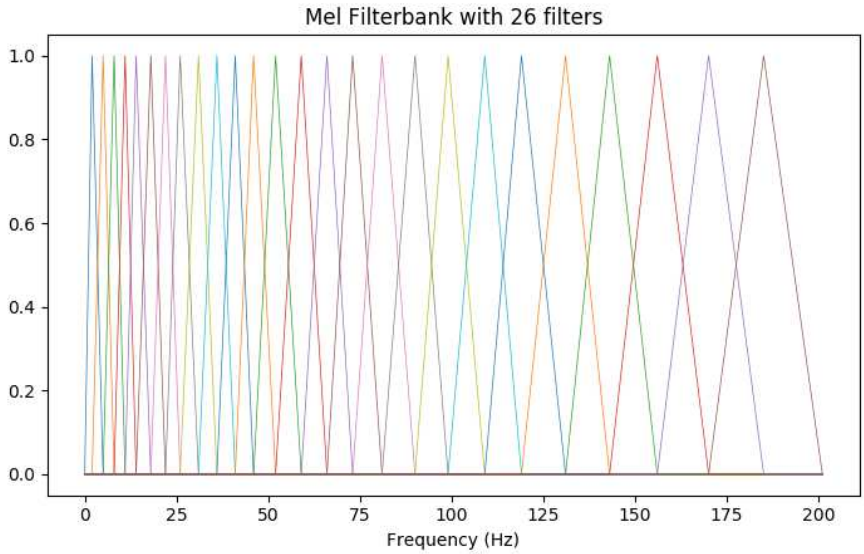

Figure 4. The mel filterbank applied to the spectrum in figure 3.

Filtering the power spectrum through the mel filterbank yielded to the mel frequency spectrum which contains energy readings that more closely represent what humans hear than the previous power spectrum. This is evident in the decreased energy readings at the higher frequencies shown in figure 5.

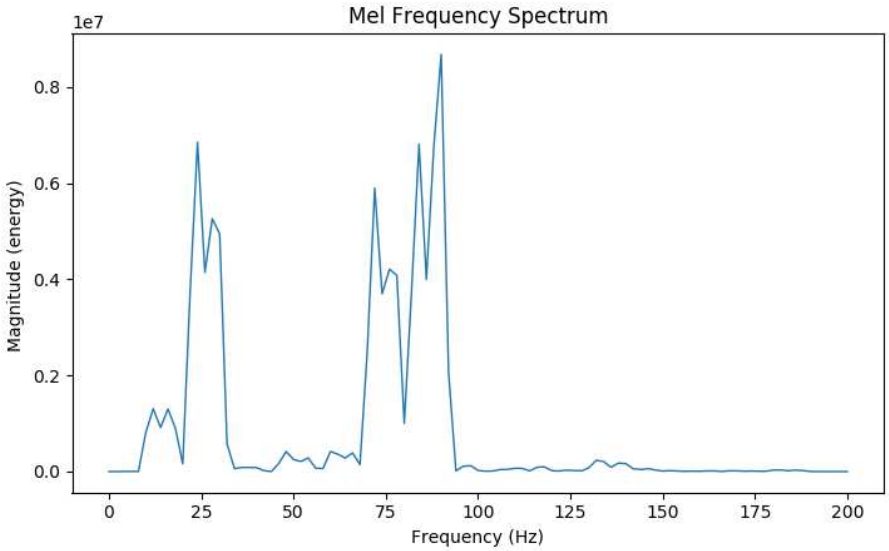

Figure 5. The mel frequency spectrum containing filterbank energies.

The mel frequency spectrum's logarithm was then computed as in figure 6. This is to further improve the features to be extracted as variation in energy levels has been proven to have little to no effect on how humans perceive sound [14]. Thus, taking the logarithm of the mel frequency spectrum still encapsulates the input signal accurately.

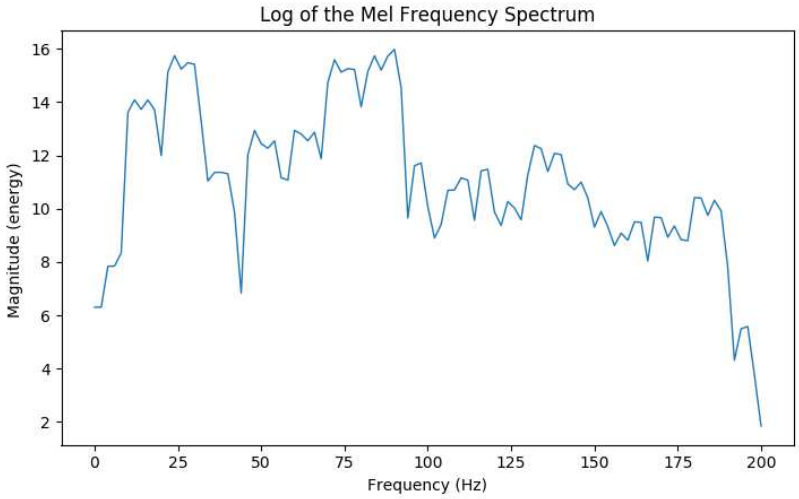

Figure 6. The logarithm of the mel frequency spectrum.

Finally, the Discrete Cosine Transform (DCT) algorithm was used on the log mel frequency values to get the MFCCs from the generated cepstrum. General implementations of MFCC extraction as in [15] result to 12 cepstral coefficients plus its energy coefficient, 13 delta cepstral coefficients, and 13 double delta or acceleration coefficients. Hence a total of 39 MFCCs can be used as features.

For this experiment, only the first 13 of the 39 MFCCs were extracted as feature vectors from each audio file; the same feature size used in the study of [16]. The vectors were then flattened by computing the mean of every feature across all frames so that each audio file can be represented as one row in the dataset. In the end, MFCC extraction resulted to a dataset of 110 files x 13 features before the ground truths of each file were appended, 110 files x 14 features after.

A summary of all the pre-processing steps done to facilitate training of the ANN is shown in figure 7.

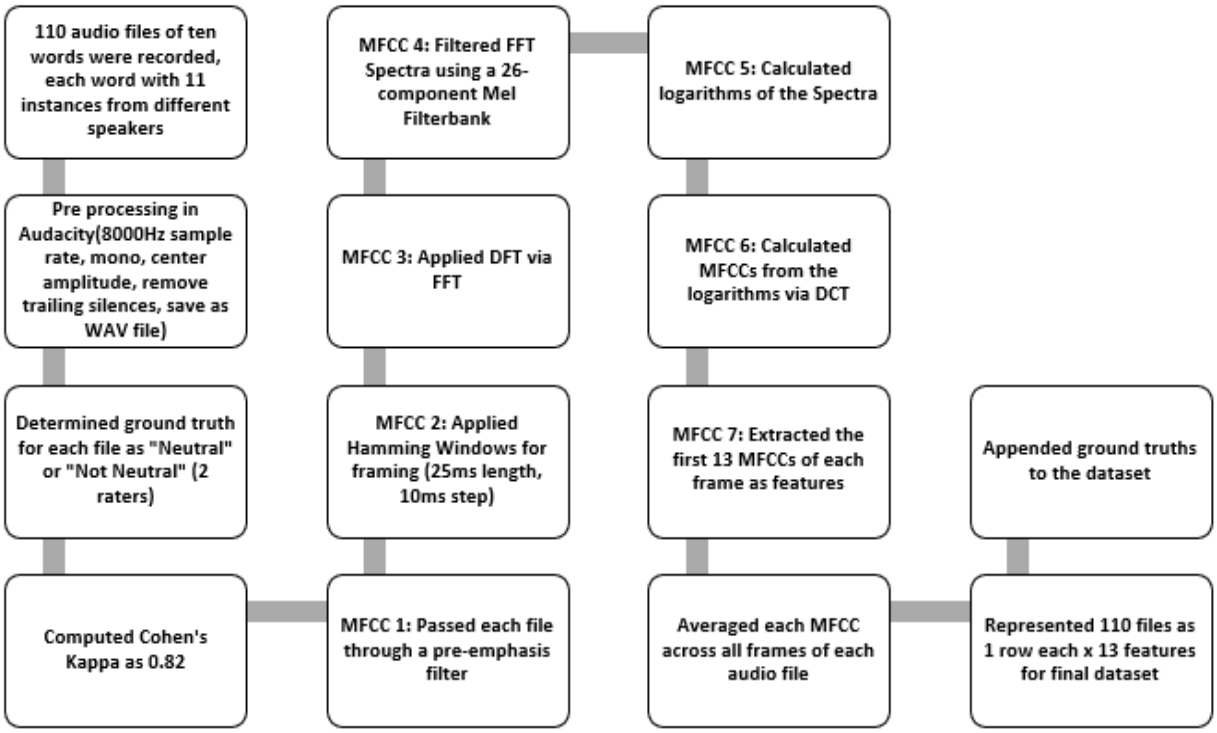

Figure 7. The pre-processing steps conducted in the study.

3.3 Training and Validation

Training and Validation was done through Sequential modelling in Keras. An ANN was created with an input layer of 13 nodes, a hidden layer of 110 nodes, a second hidden layer of 60 nodes and an output layer of 1 node as shown in figure 8.

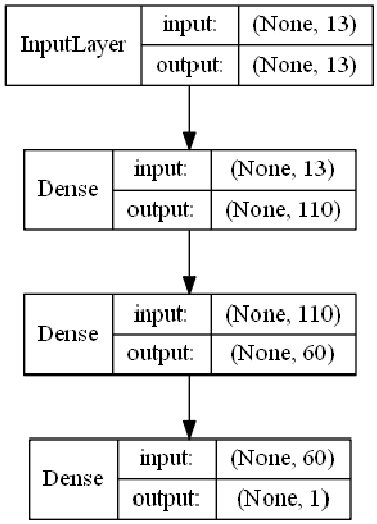

Figure 8. The ANN's architecture.

The figure describes the structure of the input and output for each layer with the notation (batch size, number of nodes). A "None" batch size as in the figure indicates that any batch size can be used during training for flexibility of experiments.

The model's structure amounts to 8,261 trainable parameters as shown in figure 9. The initial weights of each parameter were generated by the Random Normal kernel initializer in Keras. Rectified Linear Unit (ReLU) activation functions were used in the first two hidden layers of the model while a Sigmoid activation function was implemented on the output layer. The model was compiled with a Binary Cross-Entropy loss function and a Stochastic Gradient Descent optimizer.

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_1 (Dense)              (None, 110)               1540
_________________________________________________________________
dense_2 (Dense)              (None, 60)                6660
_________________________________________________________________
dense_3 (Dense)              (None, 1)                 61
=================================================================
Total params: 8,261
Trainable params: 8,261
Non-trainable params: 0
_________________________________________________________________
```

Figure 9. The total number of trainable parameters.

Stratified Ten-fold Cross Validation was used to prevent the model from overfitting during training with a batch size of 4 across 110 epochs. We also implemented a stopping condition using Keras callbacks for training to stop automatically when the minimum training loss for each fold has been reached. Accuracy of each fold was then computed and averaged to capture the ANN's overall performance. Furthermore, all misclassified files were identified for comparison with the dataset and confirm whether these files were among the files that the raters disagreed on.

4 Results and Discussion

Figure 10 shows a visualization of how the binary cross entropy loss function was minimized for both training and validation sets across all folds. Although there were noticeable fluctuations in the validation loss minimization, there were no observable deviances in the training loss minimization, and the model was still able to learn the desired weights accurately.

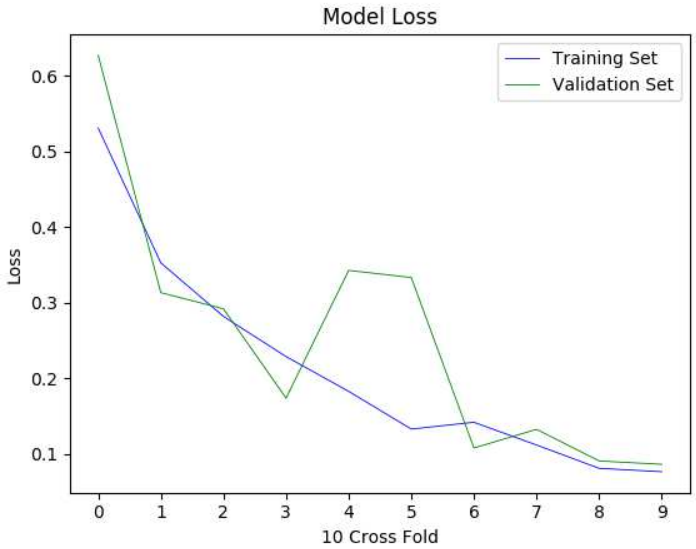

Figure 10. Binary cross entropy loss function minimization across 10 folds.

The ANN was able to correctly classify 33 out of 42 'Not Neutral' utterances and 65 out of 68 'Neutral' utterances as shown in the confusion matrix in table 2. In addition, only 1 out of the 12 misclassified files was among the utterances that had conflicting labels from the raters. This suggests that the model's errors were most likely due to the parameters not being learned properly during training and not based on the dataset's reliability.

Table 2: The ANN's confusion matrix

	Neutral	Not neutral
Neutral	33	9
Not neutral	3	65

The ANN has achieved an overall classification accuracy of 89.60% on the dataset, computed from the confusion matrix and the summary of training and validation results shown in table 3.

Table 3: Training and validation results (accuracy and loss)

Fold	Training accuracy	Validation accuracy	Training loss	Validation loss
1	0.7428	0.6625	0.5313	0.6276
2	0.8600	0.8571	0.3528	0.3134
3	0.8787	0.8484	0.2819	0.2920
4	0.9191	0.9272	0.2287	0.1737
5	0.9393	0.8636	0.1827	0.3428
6	0.9747	0.8484	0.1328	0.3333

7	0.9595	0.9696	0.1418	0.1077
8	0.9730	1.0	0.1117	0.1324
9	0.9833	0.9833	0.0808	0.0906
10	0.9775	1.0	0.07635	0.0861
Average:	0.92	0.8960		

In addition, F-Scores were also computed in every fold to back the model's reported accuracy as given by

$$H = \frac{2x_1 x_2}{x_1 + x_2}. \tag{2}$$

where H is the harmonious mean or F1 Score, x_1 is precision, and x_2 is recall. The average F-Score computed was 0.92 across all folds as shown in table 4, supporting the classifier's overall performance.

Table 4: Training and validation results (precision, recall, and f-score)

Fold	Precision	Recall	F-score
1	0.6792	0.8571	0.7504
2	0.8602	0.9183	0.8835
3	0.8101	1.0	0.8944
4	0.9250	0.9714	0.9446
5	0.8660	0.9285	0.8952
6	0.8312	0.9761	0.8940
7	1.0	0.9523	0.9743
8	1.0	1.0	1.0
9	1.0	0.9722	0.9848
10	1.0	1.0	1.0
Average:			0.9221

These findings show that the model can be expected to do well in classifying future data even without a pronunciation lexicon or dictionary used during training. In addition, the findings proved that the Stratified Ten-Fold Cross Validation can still yield good results despite the dataset having a slightly higher count of 'Neutral' utterances.

5 Conclusion

In this study, an ANN was developed to classify the neutrality of call center agents' pronunciations and develop an objective standard that a company can use for assessing their employees' or applicants' pronunciations using a small dataset of speech recordings. After ensuring the reliability of the dataset, training, and validation, the ANN achieved an accuracy of 89.60% in detecting whether utterances of 10 specific words are 'Neutral' or 'Not Neutral'. This accuracy was supported by an average F-Score of 0.92.

Therefore, the model can be expected to accurately serve as a standard that caters specifically to the call center's requirements as far as pronunciation assessment of specific words is concerned. It is important to note

however, that this performance can only be expected on new utterances that fit the context of classifying pronunciation neutrality definitive to the call center involved. Varied results may be observed if the model is tested outside of this context. Consequently, this allows for the use of the model strictly in assessing pronunciations within the call center where the dataset was collected from.

Results have also shown that the use of a standard ANN or Multilayer Perceptron for speech classification is provably effective when working with a relatively small dataset. Although a difficulty arose with the full utilization of the extracted MFCC features as standard ANNs have fixed inputs. This was addressed by flattening the MFCC vectors across all frames per individual audio file, but theoretically, other Neural Network architectures that can handle variable-length inputs and time-series data could outperform the standard ANN model and is therefore recommended for future work.

Other recommendations include creating a larger dataset with more raters for kappa computation. The use of deeper neural networks is also recommended as well as comparing the performance of different neural network architectures and feature extraction techniques.

Acknowledgements

The researchers would like to thank Jose Bien B. Tejo for facilitating the recording of audio samples from members of his team.

References

[1] Lockwood, J., Forey, G., & Price, H. (2008). English in Philippine call centers and BPO operations: Issues, opportunities and research. In *Philippine English: Linguistic and Literary* (pp. 219-241). Hong Kong University Press, HKU. doi: 10.5790/hongkong/9789622099470.003.0012

[2] Foote, J. A. and Trofimovich, P. (2017). Second language pronunciation learning: an overview of theoretical perspectives. In *The Routledge Handbook of Contemporary English Pronunciation* (pp. 93-108). doi: 10.1002/9781118346952.ch20

[3] Wotschke, I. (2014). *How Educated English Speak English: Pronunciation as Social Behaviour* (pg. 165). Frank & Timme.

[4] Copeland, J. E., Fries, P. H., Lockwood, D. G. (2000). *Functional approaches to language, culture, and cognition* (pg. 515). John Benjamins Publishing Company. doi: 10.1075/cilt.163

[5] Hu, W., Qian, Y., Soong, F. K., Wang, Y. (2015). Improved mispronunciation detection with deep neural network trained acoustic models and transfer learning based logistic regression classifiers. In *Speech Communication, Volume 67* (pp. 154-166). European Association for Signal Processing (EURASIP) and International Speech Communication Association (ISCA). doi: 10.1016/j.specom.2014.12.008

[6] Gao, Y., Xie, Y., Cao, W., Zhang J. (2015). A study on robust detection of pronunciation erroneous tendency based on deep neural network. In *INTERSPEECH-2015* (pp. 693-696).

[7] Rutherford, A. T., Peng, F., Beaufays, F. (2014). Pronunciation learning for named-entities through crowd-sourcing. In *INTERSPEECH-2014* (pp. 1148-1452).

[8] Fleiss, J. L., & Cohen, J. (1973). The equivalence of weighted kappa and the intraclass correlation coefficient as measures of reliability. *Educational and Psychological Measurement 33* (pp.613-619). doi: 10.1177/001316447303300309

[9] McHugh, M.L. (2012). Interrater reliability: the kappa statistic. *Biochemia Medica 22* (pp.276-282). doi: 10.11613/bm.2012.031

[10] Sung, Y.H. and Jurafsky, D. (2009). Hidden conditional random fields for speech recognition. *IEEE Workshop on Automatic Speech Recognition & Understanding*. doi: 10.1109/asru.2009.5373329

[11] Zheng, F., Zhang G., Song, Z. (2001). Comparison of different implementations of MFCC. *Journal of Computer Science and Technology. Volume 16, Issue 6* (pp. 582-589). doi: 10.1007/bf02943243

[12] Mecklenbrauker, W. (1989). A tutorial on non-parametric bilinear time-frequency signal representations. In *Time and Frequency representation of signals and systems* (pg.12). International Centre for Mechanical Sciences. Springer. doi: 10.1007/978-3-7091-2620-2_2

[13] Tkachenko, M., Yamshinin, A., Kotov, M., Nasatasenko, M. (2017). Speech Enhancement for Speaker Recognition Using Deep Recurrent Neural Networks. In *Lecture Notes in Computer Science* (pp. 690-696). doi: 10.1007/978-3-319-66429-3_69

[14] Malmierca, M. S., Irvine, D. R. (2005). Auditory Spectral Processing. In *International Review of Neurobiology* (pg.129). Elsevier Academic Press. doi: 10.1016/s0074-7742(05)70015-5

[15] Alshutayri, A. and Albarhamtoshy, H. (2011). Arabic Spoken Language Identification System (ASLIS): A Proposed System to Identifying Modern Standard Arabic (MSA) and Egyptian Dialect. In Communications in Computer and Information Science on (pp375 to 385). Springer. doi: 10.1007/978-3-642-25453-6_33

[16] Hirsch, H.G., Pearce, D., Deutschland, Ericsson, E. (2000). The Aurora Experimental Framework for the performance evaluation of speech recognition systems under noisy conditions. In ASR2000 - Automatic Speech Recognition: Challenges for the new Millennium Paris, France September 18-20, 2000 Proceedings.